

1-La(s) instrucción(es) necesaria(s) para cargar el dividendo 0xa30bf18a en la pareja edx:eax como

paso previo a una división sin signo son:

- b) movl \$0xa30bf18a,%eax
 xorl %edx,%edx

2-¿Qué hace gcc -O?

- a) Compilar con optimización suave

3-La diferencia entre el flag de acarreo y de overflow es que...

- c) Ambos se recalculan tras cada operación aritmético-lógica con ints, correspondiendo al programador consultar uno u otro según piense que sus datos son con o sin signo

4-De las siguientes instrucciones, ¿cuál no utiliza un modo de direccionamiento implícito?

Push

5-El direccionamiento relativo a registro base utiliza...

- c) un registro y un desplazamiento.

6-En un sistema de gestión de interrupciones mediante "polling" o sondeo, el dispositivo que

solicita la interrupción envía, junto con la señal de petición de interrupción, su correspondiente vector de interrupción.

Falso

7-Indique cuál de las siguientes afirmaciones sobre el ENIAC no es correcta:

Como los computadores actuales, era una máquina binaria, es decir, los números estaban representados en forma binaria y los cálculos aritméticos se realizaban también en el sistema binario.

8-Se puede programar un controlador de interrupciones 8259 de manera que atienda equitativamente a 8 dispositivos de igual prioridad (cada vez que se atiende a un dispositivo, éste

pasa automáticamente a tener la prioridad más baja).

Verdadero

9-¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de

64bits en el que se ha instalado también el compilador de 32bits?

-m elf_i386

10-En una bomba como las estudiadas en prácticas, del tipo...

0x080486e8 <main+120>: call 0x8048524 <strcmp>

```
0x080486ed <main+125>: test %eax,%eax  
0x080486ef <main+127>: je 0x80486f6 <main+134>  
0x080486f1 <main+129>: call 0x8048604 <boom>  
0x080486f6 <main+134>: ...
```

- a) el valor que tenga %eax
- b) el string almacenado a partir de donde apunta %eax
- c) el entero almacenado a partir de donde apunta %eax
- ✓ d) ninguna de las anteriores

11-¿Qué modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?

2

12-En IA32, el registro contador de programa se denomina:

EIP

13-La desventaja de las transferencias por bloques en un bus es que hay que transmitir todas las direcciones consecutivas de los datos del bloque.

Falso

14-La consulta de estado que se puede llevar a cabo en la E/S programada sirve para...

Consultar si el dispositivo está dispuesto para recibir datos o tiene datos disponibles

15-El controlador de interrupciones programable 8259 no permite enmascarar selectivamente líneas de interrupción.

Falso

16-El concepto de "ventanas de registros solapadas" es un mecanismo eficiente de llamada/retorno de procedimientos, utilizado en el procesador RISC-I.

Verdadero

17-Una posible codificación en microinstrucciones de la instrucción CALL X es:
 $SP=SP-1 ; m[SP]=PC ; PC=X$

18-La técnica de sondeo, escrutinio o "polling"...

Se utiliza para identificar la fuente de una interrupción

19-El lenguaje máquina es...

Difícil de codificar manualmente.

20-Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:
Compilador

21-Un puerto de entrada de un bit puede estar constituido únicamente por un simple biestable tipo
D.
Falso

22-Si el registro EAX contiene X, la sentencia en C
x &= 0x1;
se traducirá a ensamblador como:
andl \$1, %eax

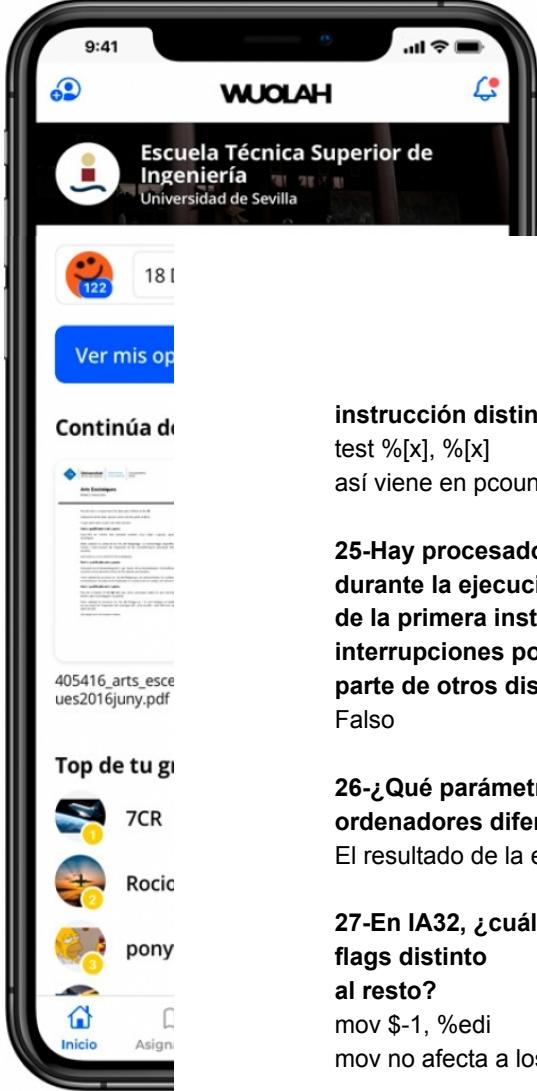
23-En la práctica "media" se suma una lista de 32 enteros de 4 B con signo para producir una media y un resto usando la instrucción IDIV. ¿Cuál de las siguientes afirmaciones es falsa?

La media se redondea al entero más próximo
división truncada IDIV - no se redondea, se trunca

24-La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:

```
int popcount3(unsigned* array,
int len){
    int i, res = 0;
    unsigned x;
    for( i=0; i<len; i++ ) {
        x = array[i];
        asm("ini3: \n"
            "shr %[x] \n"
            "adc $0, %[r] \n"
            "add $0, %[x] \n"
            "jne ini3 \n"
            : [r] "+r" (res)
            : [x] "r" (x );
    }
    return res;
}
```

Esta función produce siempre el resultado correcto, a pesar de que una instrucción máquina en la sección `asm()` es distinta a la que se esperaba después de haber estudiado `pcount_r` en teoría. La



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

instrucción distinta también se podría haber cambiado por...

test %[x], %[x]
así viene en pcount_r (Tema 2.3 tr.38)

25-Hay procesadores que desactivan automáticamente las interrupciones sólo durante la ejecución de la primera instrucción de la ISR, con lo cual es inevitable que se produzcan interrupciones por parte de otros dispositivos de forma anidada.
Falso

26-¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?

El resultado de la ejecución de un conjunto de programas de prueba.

27-En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?

mov \$-1, %edi
mov no afecta a los flags

28-El especificador de operando de una instrucción, cuando existe, es siempre una dirección de memoria o de entrada/salida
Falso

29-Una de las diferencias entre el 8086 y el 8088 es que este último no dispone de BIU ("Bus Interface Unit").
Falso

30-Respecto a registros base e índice en IA32, la excepción es que ESP no puede ser registro índice

31-La práctica "popcount" debía calcular la suma de bits de los elementos de un array. Un estudiante entrega la siguiente versión de popcount4:

```
int popcount4(unsigned* array, int len)
{
    int i, j, res = 0;
    for(i = 0; i < len; ++i) {
        unsigned x = array[i];
        int n = 0;
        do {
            n += x & 0x01010101L;
            x >>= 1;
        } while(x);
```

```
for(j = 16; j == 1; j /= 2){  
    n ^= (n >= j);  
}  
res += n & 0xff;  
}  
return res;  
}
```

Esta función `popcount4`:

produce el resultado correcto

Caso real, entregado en prácticas. La máscara está pensada para `for(j=0;j<8;j++)`. En lugar de eso,

se hace `do...while(x)`, con lo cual no se ahorran iteraciones y todo el resultado queda acumulado

en el LSB de `n`. El `for(j)` es absurdo, no itera ninguna vez. El resultado se extrae y acumula con

`n&0xFF`. Es correcto, pero no mejora la eficiencia. `popcount2` es igual de eficiente y más elegante, porque no tiene código superfluo.

32-Una forma usual de realizar el arbitraje distribuido consiste en una competición por la concesión del bus realizada por medio del envío por cada maestro peticionario de un número de arbitraje que lo identifica, de manera que un solo número "gane", y se le conceda el bus al maestro con ese número ganador.

Verdadero

33-¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?

`-m32`

34-¿Cuál de las siguientes afirmaciones es incorrecta?

- a) En el direccionamiento inmediato el dato se encuentra en la propia instrucción
- b) El direccionamiento indexado es útil para manejo de vectores.
- c) En el direccionamiento implícito no se indica la ubicación del operando
- ***d) El direccionamiento indirecto indica la dirección del operando.

INDICA UN PUNTERO AL OPERANDO

35-¿Cuál de las siguientes afirmaciones sobre los lenguajes ensambladores es falsa?

A cada sentencia le corresponde un conjunto preestablecido de instrucciones máquina.

36-La conexión de un 8086 a un sistema de memoria y E/S requiere algún circuito externo más en modo máximo que en modo mínimo.

Verdadero

37-¿Cuál de las siguientes afirmaciones es incorrecta?

- a) Las arquitecturas RISC simplifican la decodificación
- b) Las arquitecturas RISC son del tipo registro-registro.
- c) En las arquitecturas CISC hay más instrucciones que en las RISC.
- >d) El tamaño de una instrucción en lenguaje máquina siempre ocupa dos bytes en los procesadores RISC

38-En la nomenclatura del ensamblador de IA32, una cantidad de 16 bits es designada como:

Word

39-Suponga un programa residente que utiliza DMA por robo de ciclo para reproducir a través de

una tarjeta de sonido una melodía de fondo, que se encontraba en memoria, de manera simultánea

a la ejecución de un programa de usuario. La velocidad de ejecución de dicho programa de usuario

no se verá afectada por el programa residente, ya que éste utiliza DMA.

Falso

40-Una instrucción típica de entrada / salida tiene

tiene dos argumentos: un registro del procesador y una dirección de puerto de E/S claramente inspirado en el repertorio IA32 - no encuentro dónde aparece en las transparencias -

"I da A Data" en transparencia 40 es mapeado a memoria - evitar volver a poner esta pregunta en exámenes

41-Si D es un desplazamiento, RI un registro índice e I una constante apropiada, el modo de

direcciónamiento indexado con postautodecremento realiza EA = RI+D y a continuación RI = RI-I.

(EA = Effective Address)

Verdadero

42-En un computador cuya pila "crece" hacia direcciones menores se puede simular la instrucción

PUSH con una instrucción MOV con direcciónamiento indexado con preautodecremento a través del registro SP.

Verdadero

43-La instrucción ADD Rn,#3 suma el contenido del registro Rn con el de la posición de memoria 3.

Falso

44-El fragmento de código ensamblador de un microprocesador de 8 bits

**Ids IOBuf ; Apuntar puntero pila a
; ...área mem.intermedia
Idx Count ; Inicializar X-contador
poll Ida a Status; Leer estado en A
bpl poll ; Signo(A)!=1 => repetir
Ida a Data ; Leer dato en A
psh a ; Transferir dato a pila
dex ; Decrementar contador X
bne poll ; Seguir leyendo si X!=0
corresponde a:**

Entrada programada con consulta de estado

Ida a Data -> lectura del puerto de Datos

Ida a Status -> poll (consulta) de estado

45-Siendo EDX=0xf000 y ECX=0x0100, ¿cuál de las siguientes instrucciones tiene como dirección efectiva 0xf400?

xorl (%edx, %ecx, 4), %eax

46-Suponiendo que todos los registros inicialmente contienen el valor 1 y que el destino es el primer

registro, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4
mov r2, #3
add r3, r1, r1
sub r1, r3, r2
mul r3, r1, r1
5
```

47-En la captación de la instrucción:

en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción.

48-En el modo mínimo el 8086 genera menos señales que en el máximo, y por tanto depende del

controlador de bus 8288 para generar el conjunto completo de señales de control del bus.

Falso

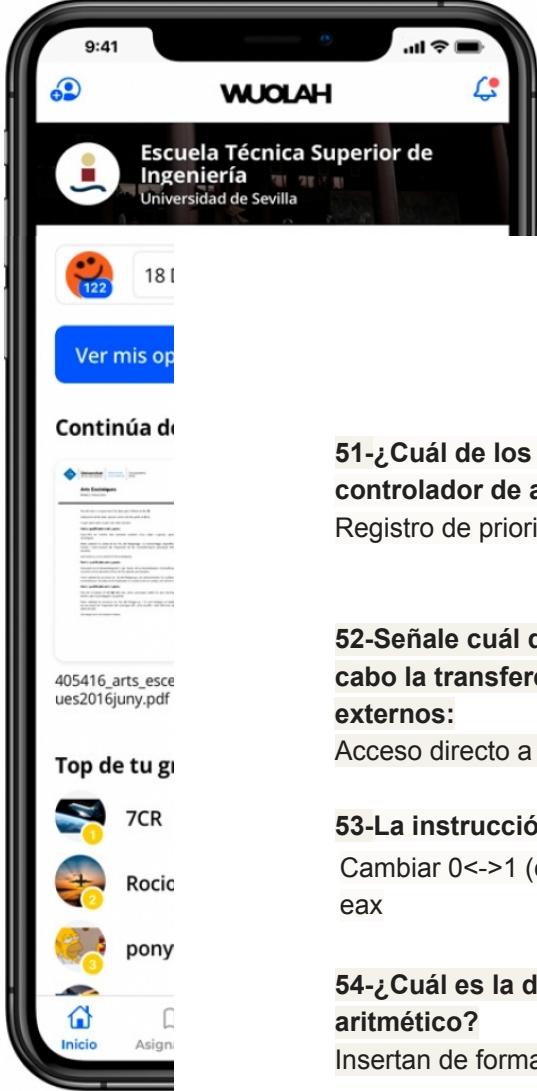
49-Si N es el número de instrucciones máquina de un programa, F es la frecuencia de reloj, y C el

número promedio de ciclos por instrucción, el tiempo de ejecución del programa será:

N·C/F

50-En general, cualquier ordenador debe incluir instrucciones específicas para E/S, por ejemplo IN y OUT.

Falso



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

51-¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?

Registro de prioridades

52-Señale cuál de las siguientes opciones es una técnica habitual para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

Acceso directo a memoria (DMA)

53-La instrucción xor \$3, %eax tiene como resultado:

Cambiar 0<->1 (complemento a 1 de) los 2 bits menos significativos del registro eax

54-¿Cuál es la diferencia entre los desplazamientos a la derecha lógico y aritmético?

Insertan de forma distinta el bit más a la izquierda

55-Cuando se ejecuta la instrucción ret al final de una subrutina:

la dirección almacenada en la cima de la pila se transfiere al contador de programa

56-Después de ejecutar el siguiente código, ¿qué variables serán igual a 0?

(Suponer ints de 32bits y longs de 64bits)

c y d

En el problema 3.4 sólo se explica que una extensión de tamaño se hace según el fuente sea signed (extensión sgn) o unsigned (ext. con ceros), pero no se explica la sección 2.2.6 (y 2.2.5) en donde se aclara que las operaciones que impliquen a algún unsigned se hacen en unsigned. Evitar esta pregunta en el futuro.

57-En el modo mínimo el 8086 genera menos señales que en el máximo, y por tanto depende del controlador de bus 8288 para generar el conjunto completo de señales de control del bus.

Falso

58-¿Es posible utilizar 4 GB de memoria en un sistema cuya CPU emplea E/S mapeada en memoria, cuyo bus de direcciones es de 32 bits y que tiene al menos un puerto de E/S? Supondremos que no se puede emplear ninguna técnica de extensión del bus de direcciones.

No

59-En un sistema de gestión de interrupciones mediante "polling" o sondeo, el dispositivo que solicita la interrupción envía, junto con la señal de petición de interrupción, su correspondiente vector de interrupción.

Falso

60-Alguna de las siguientes líneas de código sirve para definir una variable entera llamada tam en GNU/as Linux x86. ¿Cuál?

tam: .int .-msg

61-En la práctica "media" se pide calcular la media y resto de una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Qué (media : resto) se debe obtener para una lista rellena a -1 salvo el primer elemento, que valiera -31?

(-1 :-30)

62-Si declaramos int val[5]={1,5,2,1,3}; entonces

&val[2] es de tipo int* y vale lo mismo que (void*)val+8

En Sep15 faltaba (void*) y entonces sería falsa por aritmética de punteros

63-Un procesador de 8 bits, ¿a cuántos puertos de E/S podrá acceder?

Depende del método de selección de periféricos que emplee

64-Las interrupciones generadas por el teclado interrumpirán al procesador:
sólo si el procesador tiene activado el indicador de habilitación de interrupciones

65-La codificación Huffman no es la más utilizada debido a que el promedio del código de operación no es mínimo.

Falso

66-¿Qué hace gcc -O1?

Compilar con optimización

67-En un máquina con arquitectura de pila, todas las instrucciones aritméticas tienen dos operandos implícitos: la cima de la pila y el elemento siguiente de la cima de la pila.

Falso

68-Se suelen utilizar PLA en las unidades de control cableadas.

Verdadero

69-Suponga que la micropalabra de una máquina microprogramada tiene 8 bits de ancho y se usan 16 micropalabras diferentes en un microprograma de 256 micropalabras. Si se usa nanoprogramación...
se ahorran bits pero el funcionamiento es más lento.

70-De los siguientes grupos de técnicas de E/S, ¿cuáles están controlados por programa?

E/S programada y E/S mediante interrupciones.

71-¿Cuál de las siguientes instrucciones x86 se puede usar para sumar dos registros y guardar el resultado sin sobrescribir ninguno de los registros originales?

lea

72-La instrucción leave equivale a:

movl %ebp, %esp; popl %ebp

73-En el 8086, si tras una instrucción de comparación CMP A,B aparece una instrucción JC, ésta realiza un salto si A > B, siendo A y B números sin signo.

Falso

74-La instrucción necesaria para cargar 0x07 en %eax es:

movl
\$0x07,%eax

75-Compilar .c→exe (de fuente C a ejecutable) usando sólo as y ld, sin gcc...

No se puede

76-Si varios dispositivos comparten una única línea de solicitud de interrupción:

todas son ciertas

77-¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop:

push #1
push #2
push #3
pop a
push #4
pop a
pop a

78-¿Cuál de las siguientes secuencias de instrucciones calcula a=b-a, suponiendo que %eax es a y %ebx es b?

Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una

79-La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned * array, int len)
{
    int i, result = 0;
    unsigned x;
    for (i=0; i<len; i++){
        x = array[i];
        asm( "mov %[x], %%edx \n\t"
            "shr $16, %%edx \n\t"
            "shr $8, %%edx \n\t"
            "xor %%edx,%%edx \n\t"
            "setp %%dl      \n\t"
            "movzx %%dl, %[x] \n\t"
            : [x] "+r" (x)
            :
            : "edx"
        );
        result += x;
    }
    return result;
}
```

Esta función parity6:

no es correcta; fallaría por ejemplo con array={0,1,2,3}

Caso real, entregado en prácticas. Las tres primeras instrucciones asm se pierden al poner edx a 0 usando xor. Consecuentemente, se activa PF para ajustar a impar, y termina siendo x=1. Es decir, todos los elementos del array contabilizan paridad=1. El array {1,2,4,8} pasa desapercibido, pero {0,1,2,3} debería producir resultado=2<>4.

80-En el contexto del lenguaje máquina, el acrónimo ISA suele referirse a:

Instruction Set Architecture



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

81-El polling consiste en chequear cada fuente de eventos (por ejemplo interrupciones) en algún orden predeterminado.

Verdadero

82-En un microprocesador de 4 bits, una operación en la que el bit 0 de un registro se copia en el acarreo, después el bit 1 se copia en el bit 0, después el bit 2 se copia en el bit 1, y por último el bit 3 se copia en el bit 2, es:

Un desplazamiento aritmético a la derecha.

83-Alguna de las siguientes señales NO es salida de la unidad de control.

¿Cuál? Dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)

84-La segmentación de cauce...

- a) permite ejecutar varias instrucciones concurrentemente
- b) acelerar la ejecución de un programa
- c) provoca riesgos debido a datos
- d) todas las respuestas son ciertas

85-El bus AGP permite que los aceleradores gráficos accedan directamente a texturas almacenadas en la memoria principal.

Verdadero

86-¿Qué método de control de acceso directo a memoria es preferible por velocidad (más rápida), economía (coste no prohibitivo) y conveniencia de diseño (compatible con memorias y sistemas actuales)?

Transferencia de bloques o parada de CPU

87-En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32 bits (sin signo)? Se usa notación decimal y espacios como separadores de millares/millones/etc.

1 000 000 000

Se pasa, 1000 millones >> 128M

88-¿Qué método de identificación de la fuente de una interrupción suele ser más económico desde el punto de vista hardware?

La identificación mediante la técnica de sondeo

89-Un computador tiene una memoria de control de 16 Kpalabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorramos usando nanoprogramación en lugar de microprogramación?

a)

3928652

b)

259206

c)

287935

✓ •d)

ninguno de los resultados anteriores es exacto

90-Las instrucciones máquina que aparecen en segundo lugar en el análisis dinámico de uso de instrucciones son las de bifurcación.

Verdadero

91-Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.

```
struct RECORD {  
    long value2;  
    int ref_count;  
    char tag[4];  
};
```

```
struct NODE {  
    double value;  
    struct RECORD record;  
    char string[8];  
};
```

También se declara una variable global "my_node" como sigue:

```
struct NODE my_node;
```

¿Cuál es el tamaño de my_node en bytes?

32 bytes

92-La función gettimeofday() en la práctica de la "bomba digital" se utiliza para
Para cronometrar lo que tarda el usuario en introducir la contraseña

93-En un sistema Linux x86-64, ¿cuál de las siguientes expresiones es equivalente a la expresión C $(x[2] + 4)[3]$? Suponer que previamente se ha declarado int **x.

$*((*(x + 2) + 4) + 3)$
sería $x[2][4+3]$, $+4+3 == +7$ sí sería aritmética punteros

94-En las arquitecturas RISC hay...

muchos registros y pocos modos de direccionamiento.

95-En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32bits (en complemento a 2)?

0xfbff ffff

96-En el direccionamiento indirecto a través de registro, la dirección efectiva... se encuentra en un registro general del procesador.

97-En 80x86, los parámetros a las subrutinas se pueden pasar:

a)
a través de variables globales

b)
a través de los registros

c)
a través de la pila
✓ d)
todas las anteriores son ciertas

98-La etiqueta del punto de entrada a un programa ensamblador en el entorno de las prácticas 1 a 4 (GNU/as Linux x86) es:

_start

P3 y P4 se redactan en C.

P1 y P2 sí son en ensamblador GNU/as Linux x86.

Incluso en P2 se llega a usar main para ensamblar con gcc ya que usamos printf.

En cualquier caso, las otras opciones son descabelladas

99-Sean un int*a y un int n. Si el valor de %ecx es a y el valor de %edx es n, ¿cuál de los siguientes fragmentos de ensamblador se corresponde mejor con la sentencia C return a[n]?

mov (%ecx,%edx,4),%eax

ret

100-Durante un robo de ciclo DMA el procesador mantiene en alta impedancia el bus de direcciones.

Verdadero.

101-Una cola de precaptación sirve para:

Reducir el efecto de los fallos de cache

102-¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila: push #4; push #7; push #8; add; push #10; sub; mul?

20

103-En el RISC-I, una ventana de registros contiene:

registros de propósito general.

104-En un camino de datos con un solo bus, para realizar la operación de copia de un registro r1 en un registro r2, es decir $r2 \leftarrow r1$, es necesario:

Habilitar la salida triestado del registro r1 y activar la carga del registro r2

105-Se llama bit slice a la operación de desplazamiento de bits que realiza una instrucción SHL o SHR.

Falso

106-De los siguientes grupos de técnicas de E/S, ¿cuáles están controlados por programa?

E/S programada y E/S mediante interrupciones.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

107-Respecto a MBR y MAR

AR requiere menos señales de control que MBR
en Tema 3 tr.10, sólo una (LoadMAR), frente a 4 de MBR (Load/Enable Mem/Bus)

108-Las instrucciones máquina más usadas según el análisis dinámico son las de bifurcación

Falso

109-Si %edx contiene 0xf000 y %ecx contiene 0x0100, el direccionamiento 0x80(%ecx,%edx,2) se refiere a la posición

0x1e180

110-Señale cuál de las siguientes opciones no es un modo para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

E/S por flanco

111-El hecho de utilizar "polling" no implica que la prioridad de los dispositivos interruptores quede fijada mediante encadenamiento ("daisy-chain").

Verdadero

112-¿En qué pareja de registros están el dato/instrucción que se leerá o escribirá en memoria, y la dirección de memoria?

MBR y MAR

113-Debido al pequeño número de operaciones simples, la sección de control de un procesador RISC puede ser cableada en lugar de microprogramada.

Verdadero

114-Se podría diseñar una CPU microprogramada de manera que la captación y la ejecución de microinstrucciones se solapasen en el tiempo.

Verdadero

115-De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...

El bucle do-while

116-¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción IA32 add array(%ebx,4), %edx?

Direccionamiento a registro

117-La instrucción test es...

Lo mismo que and, pero no guarda el resultado, sólo ajusta los flags

118-El primer computador electrónico basaba su funcionamiento en:

tubos de vacío

1^a generación

119-¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?

En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

suponiendo convención cdecl x86, porque x86-64 usa regs.

120-¿Cuál es la característica tecnológica principal de la segunda generación de computadores?

Los transistores

121-Un procesador cuya instrucción CALL guarda la dirección de retorno en un registro:

No permite llamadas anidadas ni recursivas.

122-Un procesador de 8 bits, ¿a cuántos puertos de E/S podrá acceder?

Depende del método de selección de periféricos que emplee

123-Las instrucciones de salto...

complican el diseño eficiente de los procesadores segmentados.

124-En la técnica de salto retardado:

El compilador puede reorganizar el código para llenar los huecos de retardo con instrucciones útiles

125-En la actualidad todos los microprocesadores utilizan pipeline, tanto RISC como CISC.

Verdadero

126-En una máquina con 32 registros direccionables e instrucciones de 16 bits, es posible codificar 63 instrucciones de dos registros, 31 instrucciones de 1 registro y 32 instrucciones de 0 direcciones.

Verdadero

127-¿Qué nº de modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?

2, el modo 2.

128-En una arquitectura RISC típica:

suele usarse segmentación

129-El controlador de DMA programable 8237 puede realizar una operación de acceso directo a memoria en la que se transfiera un bloque de 512 KBytes sin intervención de la CPU una vez comenzada la transferencia.

Verdadero

130-¿Cuál de los siguientes elementos no forma parte del canal de un controlador de acceso directo a memoria?

Registro de prioridades

131-¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:

push #4

push #7

add

push #10

sub

a) 1

132-Un procesador está segmentado en las etapas F, D, E, M y W. Cada una de ellas consume un tiempo t. La aceleración ideal (si no hay riesgos) al ejecutar n instrucciones respecto a un procesador no segmentado será:

a) $5n / (4+n)$

$kn/(k+n-1)$, con

k=5

133-La idea de desarrollar máquinas CISC surgió para:

tener instrucciones cercanas al lenguaje de alto nivel.

134-¿En qué método para determinar la dirección de comienzo de una rutina de servicio de interrupción se envia parte de dicha dirección?

Direccionamiento relativo

135-Un procesador con unidad de control microprogramada tiene un generador de direcciones de memoria de microprograma en lugar de otros mecanismos de resolución de dirección efectiva.

Falso

136- Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos y un computador Y tarda 14 segundos en ejecutar ese mismo programa, ¿cuántas veces es más rápido el computador Y que el X?

1.857

137-Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%[r]"  
    :[r] "+r" (result)  
    :[i] "r" (i),  
    [a] "r" (array)  
);
```

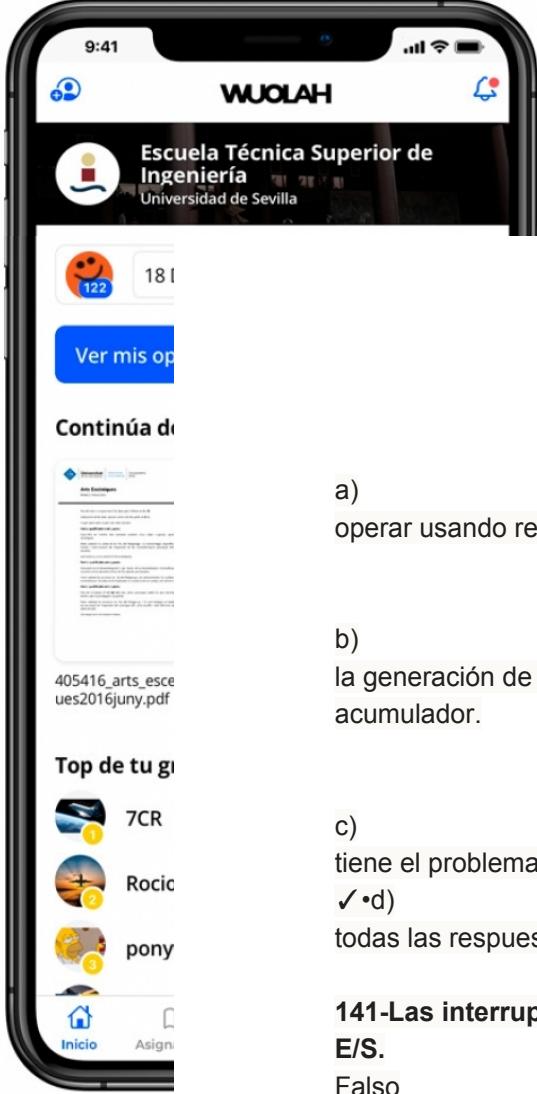
¿Cuál de las siguientes afirmaciones es correcta?

- a) r es una posición de memoria de entrada/salida
- b) el código de retorno de la función asm se fuerza a que esté en la variable result
- c) i es un registro de entrada
- d) a es una posición de memoria de entrada

138-Un computador que utilice únicamente controladores programables 8237 para realizar el DMA por robo de ciclo puede realizar:
más de cuatro transferencias por DMA concurrentes

139-La instrucción máquina DI (Disable Interrupts), conocida como CLI (Clear Interrupt Flag) en x86, se utiliza para desactivar:
Todas las interrupciones enmascarables

140-En una arquitectura de registros (a nivel de lenguaje máquina):



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

a)

operar usando registros es más rápido.

b)

la generación de código resulta más simple que en arquitecturas de pila o acumulador.

c)

tiene el problema de que las instrucciones pueden ser largas.

✓ •d)

todas las respuestas anteriores son ciertas.

141-Las interrupciones se utilizan únicamente para realizar operaciones de E/S.

Falso

142-¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C (en convención cdecl x86)?

a)

Al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

b)

Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero

•c)

Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop

X

d)

Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable

143-El objetivo del control residual es optimizar el tamaño del microprograma.

Verdadero

144-Uno de los objetivos del uso de direccionamiento relativo al contador de programa en los saltos es posibilitar la escritura de programas relocalizables.

Verdadero

145-En general, el acceso a una palabra no alineada es más lento que a una alineada.

Verdadero

146-El pipeline o segmentación consiste es una técnica para solapar la ejecución de varias instrucciones máquina.

Verdadero

147-El 8086 tiene:

14 registros de 16 bits y un bus de direcciones de 20 bits

148-En una memoria de bytes que contuviera a partir de la posición 0 los valores 1,0,0,0,0xFE,0xFF,0xFF,0xFF, se puede decir que...

a)

Hay una palabra de 16bit big-endian con valor 1 en la posición 0

b)

Hay una palabra de 16bit little-endian con valor 254 en la posición 3

c)

Hay una palabra de 32bit little-endian con valor -1 en la posición 4

d)

Todas las respuestas anteriores son incorrectas

149-Los procesadores comerciales con unidad de control microprogramada suelen almacenar los microprogramas...

en una ROM.

150-El arbitraje estático de un bus es muy simple, pero es poco flexible, y además hay que calcular previamente qué ancho de banda va a requerir cada dispositivo para no bloquear al más rápido.

Verdadero

151-Si el registro %eax contiene el siguiente valor binario:

11111111 10101010 01010101 11110000

¿Cuál será el valor de %eax tras ejecutar la instrucción xorb %al, %al?

11111111 10101010 01010101 00000000

152-¿Cuál de las siguientes tareas no es responsabilidad de un circuito de interfaz o controlador de periféricos sencillo?

Ejecutar el programa de transferencia de información entre el procesador y los dispositivos de E/S

153-¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?

a)

posición 0: FF pos.1:FF pos.2: FF pos.3: 00

b)

0:00 1:FF 2:FF 3:FF

c)

0:00 1:01 2: 00 3:80

✓ d)

Ninguna de las anteriores

154-La E/S por DMA requiere la presencia de un circuito controlador de DMA.

Verdadero

155-¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de una subrutina según la convención cdecl para IA32?

ebx

156-Una señal Bus Grant sirve para indicar que se cede el control del bus a un módulo que lo había necesitado.

Verdadero

157-La instrucción INTO del 8086 realiza la entrada de un dato desde un puerto de entrada al registro AL o AX.

Falso

158-Estudiando el listado de una función C presuntamente compilada con gcc en modo 64bit (x86-64), nos dicen que la primera instrucción, movl %eax, (%rdi), carga en EAX el valor adonde apunta el primer argumento.

Está mal, porque EAX no se carga con ningún valor

159-Alguna de las siguientes no es una operación básica de la unidad de control

a)

(leer o escribir) un registro (de / a) memoria

✓ •b)

(guardar o recuperar) un registro (en / de) la pila

c)

transferir un registro a otro

d)

realizar operación ALU y guardar resultado en registro

160-En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1 B), media palabra (2 B) y palabra (4 B), se almacenan a partir de la posición 0xCAFEBAB0 cuatro palabras con valores -1, -2, -3, -4. ¿Qué se obtendría al consultar la media palabra de la posición 0xCAFEBABE?

-1

los contenidos son

CAFEBAB0: FF FF FF FF

CAFEBAB4: FE FF FF FF

CAFEBAB8: FD FF FF FF

CAFEBABC: FC FF FF FF

las últimas dos posiciones, a partir de CAFEBABC, contienen FF FF, que es -1



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

161-Si usamos una estructura de bus con DMA:

la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.

162-La diferencia entre temporización de bus asíncrona y semisíncrona es que en la asíncrona las transferencias ocurren en algún múltiplo de ciclo de reloj, y en la semisíncrona no existe reloj del bus.

Falso

163-Una sentencia en C del tipo "while (test) body;" puede transformarse en código "goto" como:

```
if (!test) goto done;  
loop:  
    body;  
    if (test) goto loop;  
done
```

164-Alguna/s de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente es/son:

El diseño de la CPU es más sencillo.

165-Respecto a las unidades de control nanoprogramadas:

La anchura de la memoria de nanoprograma es la misma que la de memoria de micropograma en un diseño de la misma unidad de control que no usara nanoprogramación.

166-¿Cuál de los siguientes fragmentos es correcto para comenzar un programa en ensamblador que conste de un solo archivo .s?

```
.section .text  
.global _start  
_start:
```

167-Las arquitecturas memoria-memoria tienen la ventaja de emplear instrucciones máquina muy cortas y así los accesos a memoria son mínimos.

Falso

168-¿Cuál de las siguientes parejas de buses contiene sólo buses de tipo serie?

169-De las siguientes instrucciones, ¿cuál utiliza un modo de direccionamiento no implícito?

push

170-En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de

5 bits

171-En un computador cuya pila "crece" hacia direcciones menores se puede simular la instrucción PUSH con una instrucción MOV con direccionamiento indexado con preautodecremento a través del registro SP.

Verdadero

172-El registro RDM (MAR en inglés) contiene la última instrucción o dato leído de memoria o el dato que se va a escribir en memoria.

Falso

173-¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa en lenguaje C? (el fichero sin extensión es un ejecutable):

fich.c → fich.s → fich.o → fich

174-Técnicas que se pueden usar para determinar la causa de una interrupción (señalar la opción incorrecta)

a)

interrupciones vectorizadas

b)

múltiples líneas de interrupción INT1#, INT2#...

c)

consulta de estado, o polling

✓ d)

Línea de reconocimiento INTA#

175-En las instrucciones de salto condicional se suele usar direccionamiento relativo.

Verdadero

176-Para compilar un programa escrito en C en el entorno GNU/Linux se usa el programa:

gcc

177-La tendencia actual y futura en buses de periféricos es pasar de diseños serie a paralelo.

Falso

178-En el fragmento de código

```
804854e:e8 3d 06 00 00 call 8048b90 <main>
8048553:50          pushl %eax
```

la instrucción call suma al contador de programa la cantidad:

0x0000063d

179-En el pseudocódigo usado para representar las microinstrucciones, la expresión “goto f(IR)”:

Salta a una dirección de memoria de control que depende de la instrucción máquina actual.

180-¿Cuál fue el primer procesador de Intel de 64-bits en la familia x86(-64)?

Pentium 4F

181-Un procesador con una unidad de control microprogramada tiene una memoria de control de 256 palabras de 16 bits, de las que 128 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?

256 bits

182-En el secuenciamiento de microinstrucciones explícito cada microinstrucción incluye la dirección de la microinstrucción siguiente.

Verdadero

183-La instrucción JGE / JNL provoca un salto si...

OF = SF

basta recordar que "Less" no era un flag solo (es OF^SF)

recordar también que "Below" comprueba CF

184-El direccionamiento directo a memoria utiliza...

un desplazamiento.

185-En un sistema con interrupciones vectorizadas, el dispositivo o interfaz siempre suministra la dirección de la rutina de servicio de interrupción, aunque a esa dirección le falten bits.

Falso

186-El direccionamiento relativo necesita hacer uso de una tabla de direcciones, cada una de las cuales apunta a la base del siguiente dato.

Falso

187-¿Cuál de los siguientes no es un tipo de bus?

Secuencial

opuestos a buses paralelos son los buses serie

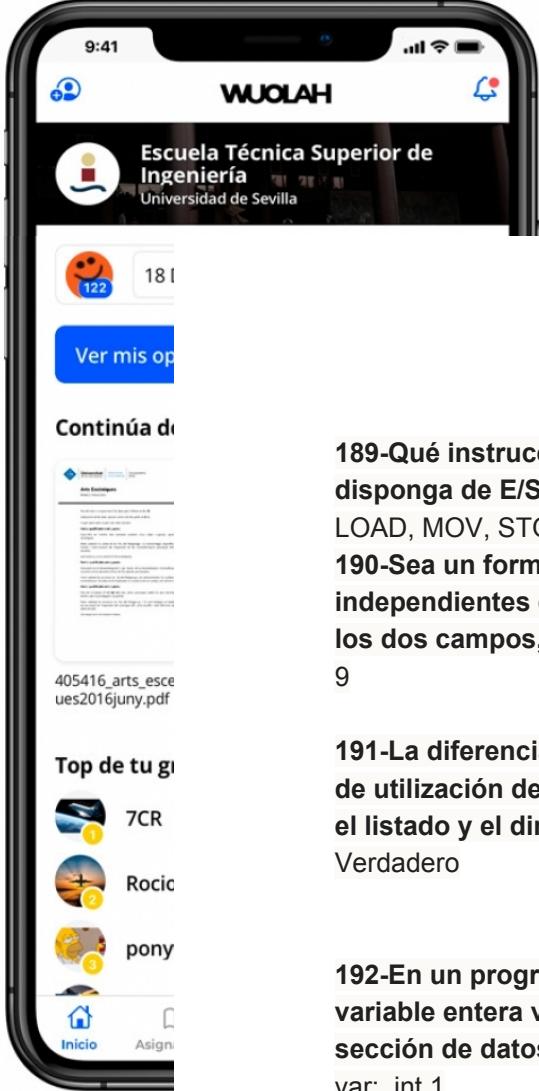
opuestos a programas secuenciales son los programas paralelos

188-La operación aritmética calculada por el programa

```
mov $5, %eax  
mov $3, %ebx  
mov $7, %ecx  
mov $8, %edx  
mul %ebx, %ecx  
add %ecx, %eax  
sub %edx, %eax
```

es:

$5 + (3 \times 7) - 8$



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

189-Qué instrucciones son típicas del repertorio de un procesador que sólo disponga de E/S mapeada en memoria?

LOAD, MOV, STORE

190-Sea un formato de microinstrucción que incluye dos campos independientes de 10 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

9

191-La diferencia entre un análisis estático y uno dinámico sobre la frecuencia de utilización de las instrucciones máquina es que el estático recuenta sobre el listado y el dinámico contabiliza las repeticiones debidas a bucles, etc.

Verdadero

192-En un programa en ensamblador queremos crear espacio para una variable entera var inicializada a 1. La línea que hemos de escribir en la sección de datos es:

var: .int 1

193-El espacio direccionable de memoria de un computador depende del diseño del:

Bus de direcciones

194-Si almacenamos según el criterio little-endian la palabra de 64 bits 0xFACEB00C a partir de la dirección 0xCAFEBAE, el byte 0xCE quedará almacenado en la dirección:

0xCAFEBAC0

195-Un fragmento de una “bomba” desensamblada es:

```
0x0804873f: call 0x8048504 <scanf>
0x08048744: mov 0x24(%esp),%edx
0x08048748: mov 0x804a044,%eax
0x0804874d: cmp %eax,%edx
0x0804874f: je 0x8048756 <main+230>
0x08048751: call 0x8048604 <boom>
0x08048756: ...
```

La contraseña/clave en este caso es...

el entero almacenado a partir de la posición de memoria 0x804a044 se hace mov 0x0804a044, %eax justo antes de cmp %eax, %edx, en donde %edx está relacionado con scanf

196-En el 8086, la secuencia de instrucciones ADD SP,10 seguida de RET es equivalente a RET 10.

Falso

197-Si la variable val está almacenada en ebx y la variable x está almacenada en eax, la sentencia val ^= x; se puede traducir a ensamblador como:
gunas de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente son:

xorl %eax,%ebx

198-La primera generación de computadores se caracteriza por el uso de:
Tubos de vacío.

199-La práctica “parity” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned* array, int len){  
    int i,j,res=0;  
    unsigned x;  
    for (i=0; i<len; i++){  
        x=array[i];  
        asm("\n"  
            "mov %[x],%%edx \n\t"  
            "shr $16, %%edx \n\t"  
            "xor %%edx,%[x] \n\t"  
            "mov %[x],%%edx \n\t"  
            "mov %%dh, %%dl \n\t"  
            "xor %%edx, %[x]\n\t"  
            "setpo %%cl \n\t"  
            "movzx %%cl, %[x]"  
            :[x] "+r" (x)
```

```
:  
    :"edx","ecx"  
);  
    res+=x;  
}  
return res;  
}
```

La sentencia asm() del listado anterior tiene las siguientes restricciones

a)

ninguna

b)

arquitectura de 32 bits

x

c)

dos entradas y una salida

d)

un registro y dos sobrescritos (clobber)

el registro es [x] "+r" y los sobrescritos son "edx","ecx"

200-En la E/S controlada por interrupciones la CPU no tiene que ejecutar un programa para realizar la transferencia de datos.

Falso

201-Hay procesadores que desactivan automáticamente las interrupciones sólo durante la ejecución de la primera instrucción de la ISR, con lo cual es inevitable que se produzcan interrupciones por parte de otros dispositivos de forma anidada

Falso

202-En los procesadores CISC gran parte del área del chip se consume en la unidad de control.

Verdadero

203-¿Cuál es la diferencia entre las instrucciones mov y lea?

mov referencia (accede) la posición indicada, mientras que lea no lo hace

204-Un sistema no segmentado tarda 10 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 5 segmentos con un ciclo de reloj de 4 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se acerca a:

2,5

205-Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.

```
struct RECORD {  
    int value2;  
    short ref_count;  
    char tag[10];  
};  
  
struct NODE {  
    long value;  
    struct RECORD record;  
    char string[8];  
};
```

También se declara una variable global "my_node" como sigue:

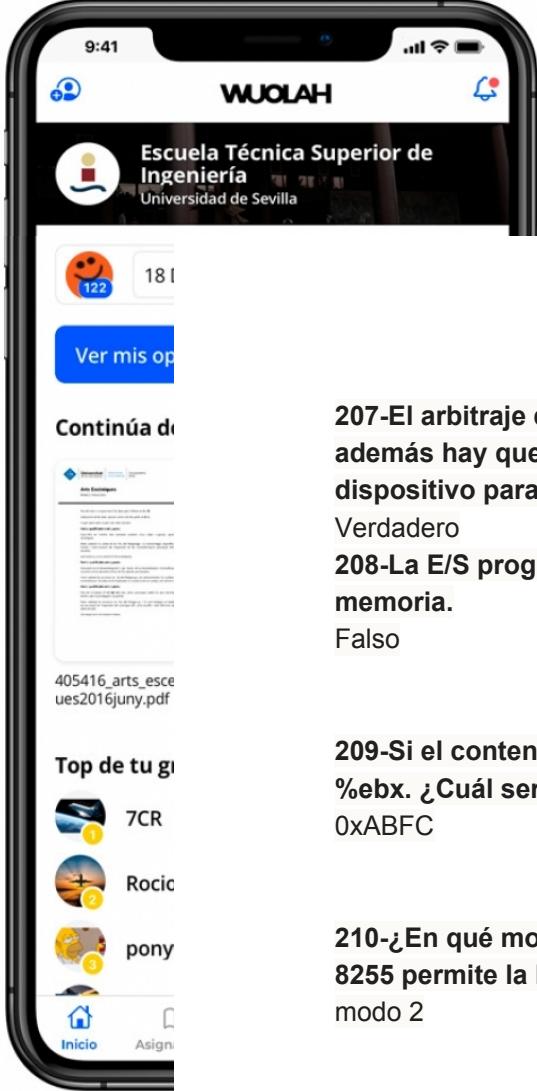
```
struct NODE my_node;
```

Si la dirección de my node es 0x600940, ¿cuál es el valor de &my node.record.tag[1]?

0x60094f

206-Un computador tiene una memoria de control de 640 palabras de 70 bits, de las que 280 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación en lugar de microprogramación?

19440



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

207-El arbitraje dinámico de un bus es muy simple, pero es poco flexible, y además hay que calcular previamente qué ancho de banda va a requerir cada dispositivo para no bloquear al más rápido

Verdadero

208-La E/S programada con consulta de estado ha de ser mapeada en memoria.

Falso

209-Si el contenido de ESP es 0xAC00 antes de ejecutar la instrucción push %ebx. ¿Cuál será su contenido tras ejecutarla?

0xABFC

210-¿En qué modo de funcionamiento la interfaz de periféricos programable 8255 permite la E/S con bus bidireccional?

modo 2

211-Utilizando E/S programada y como modo de direccionamiento selección lineal, ¿cuántos periféricos podrían conectarse a un 8086? Recordar que el 8086 disponía de E/S separada, con bus de direcciones de 20 bits para memoria y de 16 bits para E/S.

16 periféricos

212-Un sistema no segmentado tarda 50 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce ("pipeline") de 5 segmentos con un ciclo de reloj de 50 ns. Cuando se procesan muchas tareas, la ganancia de velocidad que se obtiene se acerca a 5.

Falso

213- ¿Es cierto que la consulta de estado permite averiguar el estado de un periférico y alterar la prioridad de los periféricos?

Sí, permite averiguar el estado de un periférico y también alterar la prioridad de los periféricos

214-Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA32 como en x86-64. Sólo una de ellas es FALSA. ¿Cuál?

a)

El tamaño de un puntero es 4 bytes

b)

El tamaño de un int es 4 bytes

c)

El tamaño de un double es 8 bytes

d)

El tamaño de un short es 2 bytes

215-Un computador que utilice únicamente controladores programables 8237 para realizar el DMA puede realizar más de cuatro transferencias por DMA concurrentes.

Verdadero

216-La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...

dos multiplexores de 2 a 1

217-¿Cuál de los siguientes grupos de instrucciones IA32 sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

CMP, TEST

218-¿Cuál de los siguientes grupos de instrucciones podrá pertenecer a un procesador con E/S mapeada en memoria?

LOAD, MOV, STORE

219-Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/IA-32 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como: -k(%ebp), siendo k un número constante positivo relativamente pequeño

220-¿En qué técnica para determinar la dirección de comienzo de la rutina de servicio de interrupción se fija dicha dirección en los circuitos de la CPU?
Direcciones fijas.

221-Si R0=2, R1=5 y M[3]=3 ¿Qué valor toman R0, R1 y M[3] tras ejecutarse la instrucción XOR 1h[R0],R1?
R0=2 , R1=5 , M[3]=6

222-El sufijo l de la instrucción movl significa:

Que la instrucción trabaja con operandos de 32 bits (long word).

223-Las ventanas de registros solapadas se utilizan para ahorrar registros en el diseño del procesador.

Falso

224-¿Cuál de los siguientes lenguajes no permite el paso de parámetros por referencia?

C

225-Teniendo en cuenta que el tiempo de ejecución de un programa es proporcional al producto número de instrucciones del programa * número medio de ciclos de reloj por instrucción, la filosofía CISC pretende reducir el tiempo de ejecución reduciendo el número de instrucciones del programa y la RISC reduciendo el número medio de ciclos de reloj por instrucción.

Verdadero

226-En el 80x86, la instrucción "JNE etiqueta" es equivalente a "JNZ etiqueta".

Verdadero

227-Tras comparar números con signo, para saltar si menor se utilizan:

El overflow y el signo

228-Dada la siguiente declaración en lenguaje C, una estructura de este tipo podría ocupar, bien sea en un sistema Linux IA32 o bien en uno x86-64, un total de...

```
struct a{  
    int i;  
    double d;  
    char c;  
    short s; };
```

24 B

4 B relleno entre i y d, 1B entre c y s, 4B relleno tras s.

229-Cuál de las siguientes características es posterior a la segunda generación de computadores?

RISC.

230-Si usamos una estructura de bus con DMA:

la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.

231-La especificación de un bus a nivel eléctrico debe incluir las siguientes partes:

Alimentación, impedancia, nivel de señal,...

232-¿Cuál de las siguientes afirmaciones es cierta?

En E/S independiente, las instrucciones de acceso a memoria suelen ser más largas que las de E/S

233-Un conjunto de microoperaciones compatibles constituye una microinstrucción, y una secuencia de microinstrucciones es un micropograma.

Verdadero

234-¿Cuáles de las siguientes instrucciones utilizan sólo direccionamiento implícito?

lahf, movs

235-La instrucción cmovb %edx, %eax

copia en %eax el contenido de %edx si el indicador de acarreo es 1
"below" equivale a CF



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

236- Los microprocesadores RISC puede ser implementados en un tiempo más corto, pero requieren mayor área de silicio que los CISC.
falsa

237-Una línea de interrupción organizada en colector abierto permite conectar varios dispositivos de manera sencilla sin necesidad de circuitos combinacionales (puertas OR).

Verdadera

238-En el direccionamiento directo, la dirección efectiva se encuentra en un registro general del procesador.

falsa

239-El control residual se utiliza para:

reducir el tamaño de la memoria de control

240-¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?

a) Registro contador

b) Registro de órdenes

c) Registro de dirección

**d) Registro de prioridades

241-En la E/S mapeada en memoria, el procesador (CPU) no distingue entre accesos a memoria y accesos a los dispositivos de E/S.
veradero

242-Si varios dispositivos comparten una única línea de solicitud de interrupción se puede usar "polling" para identificar el origen de una interrupción.

veradero

243-Una operación de transferencia READ-MODIFY-WRITE es ideal para implementar la instrucción TEST AND SET.

veradero

244-La convención de llamada Linux/GCC x86-32 considera, respecto a convenios de uso de registros:

3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales

245-¿Qué valor contendrá el registro edx tras ejecutar las dos instrucciones siguientes?

movl \$-1, %edx

movb \$1, %dl

11111111 11111111 11111111 00000001

246-En la realización de la práctica de la bomba digital, una parte del código máquina es el siguiente:

0x080486e8 <main+120>: call 0x8048524 <strcmp>

0x080486ed <main+125>: test %eax,%eax

0x080486ef <main+127>: je 0x80486f6 <main+134>

0x080486f1 <main+129>: call 0x8048604 <boom>

¿Cuál de los siguientes comandos cambiaría el salto condicional por un salto incondicional?

set *(char*)0x080486ef=0xeb

247-El direccionamiento inmediato no es más lento que el direccionamiento directo

verdadero

248-Supongamos dos CPU con bus de direcciones de ancho idéntico. Si una de ellas emplea E/S independiente y la otra mapeada en memoria, ¿cuál podrá acceder a una mayor cantidad de memoria?

La CPU con E/S independiente.

249-Sea un formato de microinstrucción que incluye dos campos independientes de 9 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

8

250- ¿Cuál de las siguientes afirmaciones es incorrecta?

a) El repertorio de instrucciones es el conjunto de operaciones que es capaz de interpretar la unidad de control.

b) El repertorio de instrucciones debe ser capaz de realizar una tarea en un tiempo finito.

c) El modo de direccionamiento permite determinar un operando o la ubicación del operando.

***d) Los operandos siempre están almacenados en memoria.

251- ¿De qué depende el tamaño del contador de programa?

a) de la longitud del código de operación

b) del ancho del bus de datos

c) el tamaño no importa

***d) ninguna de las anteriores es cierta

252- El número -12 se almacenará en complemento a 2 en el registro %eax como:

0xFFFFFFFF4

253- En el controlador de DMA 8237 los registros de dirección de memoria y contador de bytes están duplicados para cada canal para que en el modo de autoinicio el circuito sea capaz de recordar los valores originales de esos registros.

Verdadero

254- Un sistema no segmentado tarda 200 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce segmentado de 20 etapas con un ciclo de reloj de 12 ns. Cuando se procesan muchas instrucciones, la máxima ganancia de velocidad que podría obtenerse se acerca a:

16,67

255- El punto de entrada de un programa ensamblador en GNU/as Linux x86 se llama

_start

256- Como parte del proceso de compilación de una aplicación en lenguaje C, enlazar .o → .exe (de objeto proveniente de fuente C a ejecutable) usando sólo as y ld, sin gcc...

Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C

257- En los modos de direccionamiento del tipo Desplazamiento (Base, Índice, Factor Escala), puede usarse como desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una

variable, por su dirección)

258- Un sistema no segmentado tarda 10 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 5 segmentos con un ciclo de reloj de 4 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:

2,5

259-Indique cuál de las siguientes características no es cierta en el direccionamiento indirecto a memoria a través de memoria:

La instrucción contiene la dirección de memoria exacta en que se encuentra el objeto.

260- En la práctica “media” se pide sumar una lista de enteros *con* signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en los *dos* primeros elementos de la lista causaría overflow con 32 bits al realizar la suma de *esos dos* primeros elementos de la lista?

0x4000 0000

261- Las interrupciones no se utilizan únicamente para realizar operaciones de E/S.

Verdadero

262- Si una pila crece hacia direcciones de memoria decrecientes, una instrucción PUSH utiliza direccionamiento indexado con postautodecremento, siendo SP el registro índice.

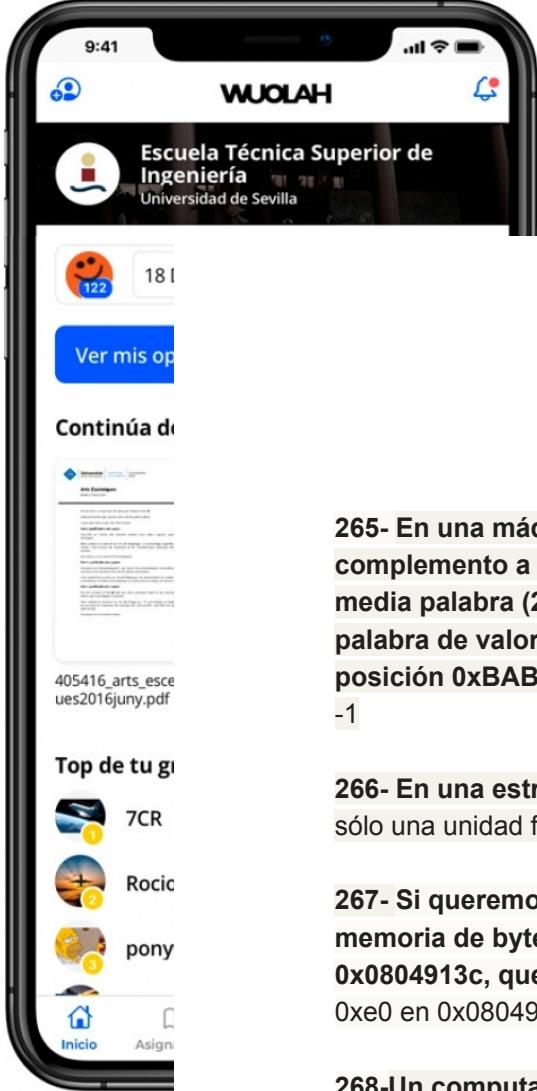
falsa

263- La técnica de sondeo, escrutinio o "polling"...

- a)Se utiliza para identificar el destino de una interrupción
- ***b)Permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos
- c)En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción
- d)Todas las respuestas anteriores son falsas

264- Cuál de las siguientes características es típica de la microprogramación horizontal?

Ninguna o escasa codificación.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

265- En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1B), media palabra (2B) y palabra (4B), si se almacena en la posición 0xBABC una palabra de valor -2, ¿qué se obtendría al consultar la media palabra en la posición 0xBABE?

-1

266- En una estructura de computador de bus único (bus del sistema): sólo una unidad funcional puede tener el control del bus en cada momento

267- Si queremos almacenar la palabra de 64bits 0x0000001f ffffffe0 en una memoria de bytes según la convención little-endian a partir de la posición 0x0804913c, quedará

0xe0 en 0x0804913c y 0x1f en 0x08049140

268- Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria. Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de 2^{10} , ¿qué tamaño de memoria admite el computador?

2^{12} palabras

269- La segunda generación de computadores se caracteriza tecnológicamente por la utilización del transistor.

V

270- Son funciones de la unidad de control:

el secuenciamiento de las instrucciones máquina

271- La ecuación básica de rendimiento calcula

cuánto tiempo tarda en ejecutarse un programa concreto conociendo su número de instrucciones y el número de etapas (promedio) y la frecuencia del procesador

272- ¿Cuál de las siguientes afirmaciones es cierta?

- a) la unidad de control necesita como entrada el registro contador de programa para saber cuál es la instrucción que debe ejecutar a continuación realmente la UC copia PC en MAR, y lo que lee en MDR (captación) lo lleva a IR que sí es entrada a la UC (decodificación)
- b) la arquitectura von Neumann de los computadores tradicionales consiste en tener almacenados los datos separados de las instrucciones en memorias distintas vimos en [T4.2EjmSeg] tr.12 que tener memoria de instrucciones separada de la de datos es una variante llamada "arquitectura Harvard", llamándose "arquitectura Princeton" la original
https://en.wikipedia.org/wiki/Von_Neumann_architecture
https://en.wikipedia.org/wiki/Modified_Harvard_architecture
- ✓ • c) el registro de estado (flags) es un registro de propósito específico cuyo contenido puede ser visto directa o indirectamente por el usuario mediante el uso de ciertas instrucciones específicas aunque no hemos visto pushf y popf, por eliminación ésta es la respuesta
- d) el registro de direcciones de memoria es un registro de propósito general que puede contener tanto direcciones como datos
MAR no está disponible al programador, lo usa la UC para indicar la dirección de memoria a leer/escribir

273- La 5^a generación de computadores comenzó aproximadamente en 1971 con el primer microprocesador.

F

274-¿Cuántos bits hacen falta como mínimo para implementar tres niveles de inhibición de interrupciones (general, nivel y máscara) en un sistema con cuatro niveles de interrupción?

7

275-Un circuito controlador de interrupciones permite resolver prioridades entre las fuentes de interrupción conectadas a él.

Verdadero

276-En un sistema de 32bits, ¿cuál de las siguientes expresiones C es equivalente a la expresión $(x[2] + 4)[3]$?

(Asumir que x se ha declarado como int **x. Recordar que C usa aritmética de punteros. Notar que muchos de los paréntesis no son necesarios, sólo se han añadido para evitar confusiones por precedencia de operadores)

$\ast(\ast(x + 2)) + 7$

277-Alguna de las siguientes técnicas NO es de utilidad para determinar la causa de una interrupción

a)

Línea de reconocimiento INTA#

b)

Interrupciones vectorizadas

x

c)

Múltiples líneas de interrupción INT1#, INT2#...

f

d)

Consulta de estado, o polling

278-¿A qué tipo de interrupciones corresponde la forma de determinar la dirección de comienzo de una rutina de interrupción en la que se envía una instrucción de bifurcación completa?

Interrupciones vectorizadas

279-El programa ensamblador siempre resuelve todas las referencias usadas en el fichero fuente.

Falso

280-La instrucción TEST_AND_SET comprueba si la posición de memoria especificada como operando vale 0 y a continuación la pone a 1, todo ello sin posibilidad de interrupción.

Verdadero

281-En una arquitectura de registros de propósito general (a nivel de lenguaje máquina):

a)

operar usando registros es más rápido.

b)

la generación de código resulta más simple que en arquitecturas de pila o acumulador.

c)

se evita el cuello de botella (por ejemplo, pila, o acumulador) que otras arquitecturas presentan al evaluar expresiones aritméticas complejas

✓ d)

todas las respuestas anteriores son ciertas.

**282-¿Cuál de las siguientes direcciones NO está alineada a double (8-byte)?
(Al no poder escribir el 2 como subíndice, aclaramos que ")2" indica binario)**

1110110101110100)2



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

283-Considerar la declaración C

`int array[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};`

Suponer que el compilador tiene la dirección de array en el registro %ecx.

¿Cómo se movería el valor array[3] al registro %eax? Asumir que %ebx es 3.

`movl (%ecx,%ebx,4),%eax`

284-Para traducir una construcción if-then-else de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente

un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional

285-Según el convenio little-endian, encontraremos el byte más significativo en una dirección menor.

Falso

286-Para conectar las salidas de dos registros hacia un bus común en el datapath...

se pueden usar dos buffers triestado.

287-Las operaciones elementales de transferencia de datos en E/S por interrupciones deben ser realizadas por programa.

Verdadero

288-Cuanto más horizontal es la microprogramación más largas son las microinstrucciones.

Verdadero

289-¿Qué técnica de E/S requiere menos atención por parte del procesador?

E/S mediante acceso directo a memoria

290-¿Qué técnica de E/S consume menos tiempo del procesador?

E/S mediante DMA

291-Una instrucción máquina puede desglosarse en las siguientes operaciones elementales:

sp := sp - 1; m[sp] := pc; pc := x

llamada a subrutina

292-EI 8255 es un interfase paralelo de periféricos que puede utilizarse para E/S programada, pero también puede generar interrupciones.

Verdadero

293-En 8086, los parámetros a las subrutinas se pueden pasar:

a)

a través de variables globales

b)

a través de los registros

c)

a través de la pila

✓ d)

todas las anteriores son ciertas

294-En arbitraje por "daisy-chain" la señal BUS_REQUEST es compartida por los maestros potenciales.

Verdadero

295-Las instrucciones JB y JNAE del Pentium provocan un salto si...

CF == 1

296-Una instrucción de llamada a subrutina es más compleja en general que una de salto.

Verdadero

297-¿Cuál de las siguientes afirmaciones acerca del daisy-chain es cierta?

a) Es
incompatibl
e con la
técnica de
sondeo o
polling

- b) Todos los componentes se conectan directamente y con igual prioridad al procesador o gestor de interrupciones
- c) Los componentes se comportan de forma cooperativa: sólo al de mayor prioridad se le concede la interrupción o se apodera del bus de comunicaciones
lo del "bus de comunicaciones" no se explica en las transparencias actuales - el daisy-chain se puede usar también como método de arbitraje para conceder el bus - las señales suelen llamarse BR/BG (Bus request/grant) en lugar de INTR/INTA (IRQ request/acknowledge)
- d) Los componentes están conectados todos con todos y un gestor centralizado decide la prioridad

298-¿En qué modo del transferencia el controlador de DMA programable 8237 no tiene en cuenta el registro contador?

Transferencia por demanda

299-¿Cuál afirmación es FALSA en arquitecturas x86-64?

- a) El tamaño de las posiciones de memoria es 64 bits ←Esta es la falsa
- b) El tamaño de un double es 64 bits
- c) El tamaño de los registros es 64 bits
- d) El tamaño de un puntero es 64 bits

300-En la práctica de la bomba, el tercer ejercicio consistía en usar un editor hexadecimal para crear un ejecutable sin “explosiones”. Para saber qué contenidos del fichero hay que modificar, se puede utilizar... (marcar la opción *falsa*)

hexedit

el enunciado implícitamente indica que no basta con hexedit. Aún así hubo quien la dejó en blanco (pocos) y quien falló (50%)

301-En los computadores con arquitectura Von Neumann los programas están en memoria principal (o caché) cuando se ejecutan.

Verdadero

302-La instrucción seta %al (seta significa "set if above"):

Pone AL a 1 si CF=0 y ZF=0

303-¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción no es cierta?

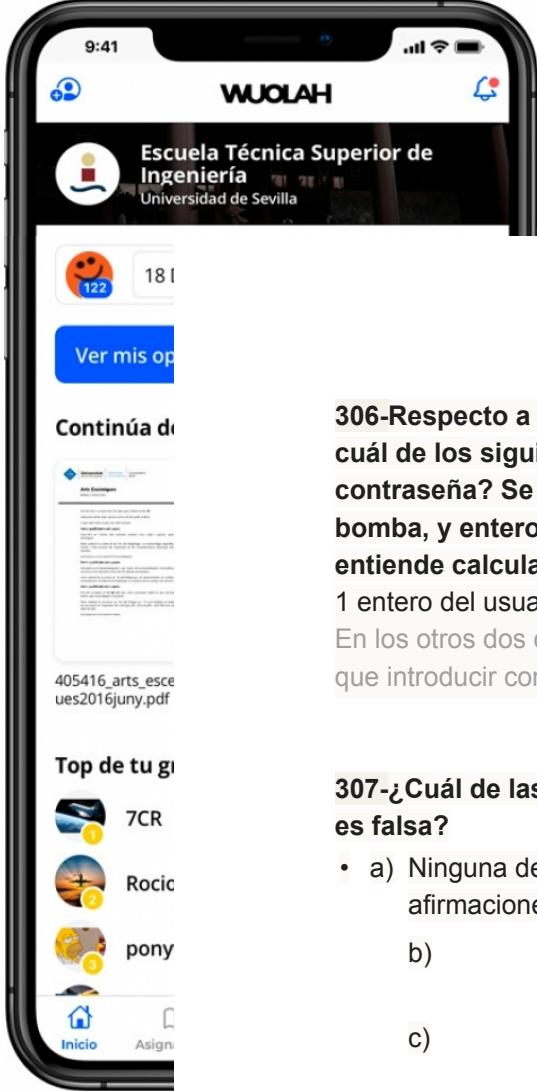
- a) Es una bifurcación normalmente externa al programa en ejecución
- b) Su objetivo es reclamar la atención del procesador
- c) Solicita que se ejecute un programa específico para tratarla
-
- d) Ninguna de las anteriores <-Esta

304-En las instrucciones aritméticas con dos operandos de un procesador con arquitectura de pila, los dos operandos, implícitos, son la cima de la pila y el elemento siguiente de la cima de la pila.

Verdadero

305-La principal desventaja de la nanoprogramación es que requiere un programa ensamblador muy complejo.

Falso



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

306-Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la contraseña? Se distingue entre enteros definidos en el código fuente de la bomba, y enteros solicitados al usuario mediante scanf(). Por "procesar" se entiende calcular el n-ésimo elemento de la serie de Fibonacci.

1 entero del usuario se procesa, y se compara con el entero del fuente

En los otros dos casos usando el debugger se puede ver el resultado literal que hay que introducir como contraseña, parando justo antes de comparar

307-¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es falsa?

- a) Ninguna de las afirmaciones es falsa <-ESTA
- b) Solicita que se ejecute un programa específico para tratarla
- c) Su objetivo es reclamar la atención del procesador
- d) Es una bifurcación normalmente externa al programa en ejecución

308-Respecto a la convención de llamada usada en Linux/gcc:

Hay registros que pueden ser modificados libremente por las subrutinas, y otros que, si se modifican, se deben restaurar posteriormente. Y también hay registros especiales

309-¿Cuál de las siguientes afirmaciones es cierta?

- a) Ninguna de las otras respuestas es cierta
- b) Los estándares ATA/IDE, SCSI y Firewire (IEEE 1394) definen buses de funcionamiento paralelo a diferencia del estándar USB que define un bus de funcionamiento serie
- c) El bus AGP se utiliza para conectar tarjetas gráficas y controladoras de disco
- d) Los buses PC XT, AT/ISA, MCA, EISA, VLB, PCI y AGP son buses de placa madre de PC <-ESTA

310-Un 8086 necesita al menos 1 KB de memoria RAM para funcionar correctamente.

Falso

311-¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC es cierta?

- a) Ninguna de las otras respuestas es cierta **<-ESTA**
- b) No todas las interrupciones se pueden generar por software
- c) Cada vector de interrupción es una palabra de 16 bits
- d) Existen 1024 vectores de interrupción

312-Respecto a la diferencia entre dispositivos activos y pasivos en un bus podemos decir:

- a) Los dispositivos pasivos siempre son esclavos
- ✓ •
- b) Los dispositivos activos pueden actuar como maestros
 - c) Las respuestas a y b son ciertas
 - d) Las respuestas a y b son falsas

**313-Respecto a registros de propósito general (RPG), el 80386 tiene:
8 registros de 32 bits**

314-Suponga una memoria de microprograma de n microinstrucciones de w bits cada una. Suponiendo que de esas n microinstrucciones, hay 2^m distintas, el ahorro en bits si se utiliza nanoprogramación es $n \cdot w - (n \cdot m + 2^m \cdot w)$.

✓

315- Sobre las técnicas de transferencia en operaciones de E/S:

- a) Pueden ser controladas por programa o por hardware
- b) Si se emplea E/S programada puede hacerse con consulta de estado o sin consulta de estado
- c) En el caso de utilizar E/S mediante DMA hace falta emplear un controlador de DMA
- d) Todas las respuestas anteriores son ciertas

316-Un 8086 puede funcionar correctamente con 1 MB de memoria ROM.

V

317-En IA32, tras dividir 0x640000 (64 bits) entre 0x8000 (32 bits), el resultado será:

0xC8 en EAX y 0x0 en EDX

318-La práctica “parity” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity5:

```
int parity5(unsigned* array, int len){
    int i,j,res=0;
    unsigned x;
    for (i=0; i<len; i++){
        x=array[i];
        for (j=sizeof(unsigned)*4;
             j>0; j=j/2){
            x^=x>>j;
        }
        x = x & 0x1;
        res+=x;
    }
    return res;
}
```

Esta función parity5:

produce siempre el resultado correcto

solemos escribir res+=x&0x1, en lugar de ponerlo en 2 sentencias C

319-El registro MBR...

contiene el valor que va a ser almacenado en la memoria, o bien se usa para recibir un valor procedente de la memoria

320-Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:

4

321-Si queremos almacenar la palabra de 16 bits 8965h en memoria según "little-endian", quedará almacenada a partir de la posición 8600h como:
en el byte 8600h se guarda 65h y en el byte 8601h se guarda 89h

322-En las últimas generaciones de computadores la mejora de prestaciones viene dada por:

avances en tecnología y avances en la estructura y arquitectura del computador.

323-Los arrays bidimensionales en lenguaje C se almacenan en orden... mayor-de-fila" (row-major)

324-La secuencialidad es la forma de direccionamiento implícita en el flujo de ejecución de un programa.

v

325-Un controlador de DMA suele ser programado con la siguiente información relativa a una operación de E/S:
tipo de operación, tamaño de bloque a transferir, dirección inicial de memoria

326-Una implementación diferencial del bus SCSI permite mayor distancia o velocidad que una "single-ended".

v

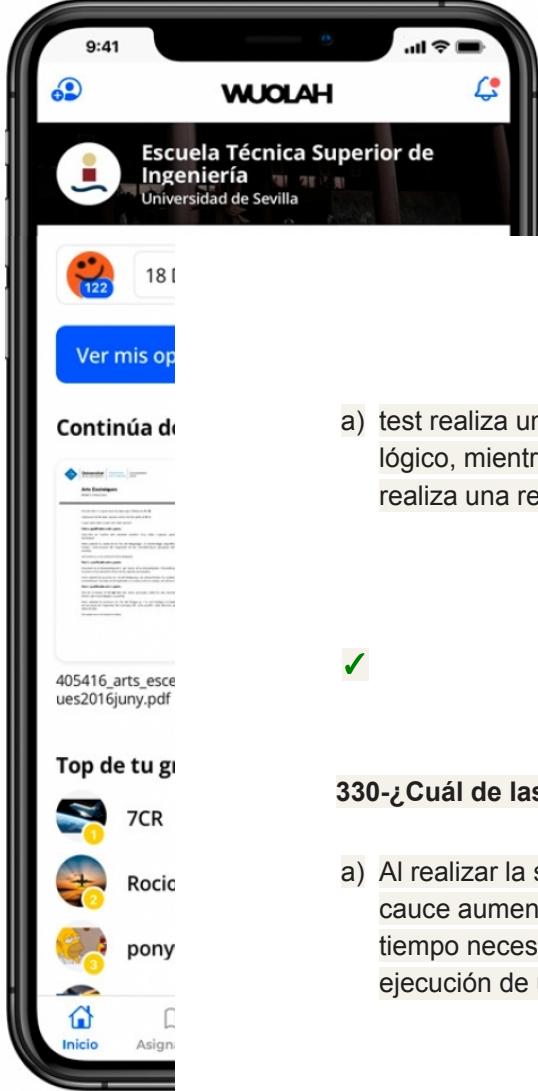
327-Más de la cuarta parte de las instrucciones máquina que se ejecutan en un programa típico son de movimiento o transferencia de datos.

v

328-A qué tipo de interrupciones pertenecen las condiciones de tiempo real y los fallos hardware?

No enmascarables

329-La diferencia entre las instrucciones test y cmp consiste en que



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- a) test realiza una operación and lógico, mientras que cmp realiza una resta



- b) test modifica sólo los flags lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
- c) ambas respuestas son correctas
- d) ambas respuestas son incorrectas

330-¿Cuál de las siguientes afirmaciones es cierta?

- a) Al realizar la segmentación de cauce aumenta en general el tiempo necesario para la ejecución de un programa



- b) Debido a que pueden existir dependencia de datos, los resultados de un programa pueden ser diferentes a si el programa se ejecutara sin segmentación
- c) La segmentación de cauce disminuye el número de instrucciones necesarias para la ejecución de un programa
- d) Ninguna de las afirmaciones anteriores

331-Una instrucción de salto si menor, para números positivos sin signo, tiene que comprobar el valor de: el bit de acarreo

332-¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:

push #4
push #7
add
push #10
sub
1

333-Para traducir una construcción if-then-else de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente

un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional

334-No en todas las instrucciones máquina
hay una fase de captura de operandos

335-El programa RISC

ld r4,(r2)
ld r5,(r3)
add r6,r4,r5
st (r1),r6

Almacena la suma de los contenidos de las posiciones de memoria direccionadas por los registros r2 y r3, y almacena el resultado en la posición de memoria direccionada por el registro r1.

336-¿Cuál de las siguientes características es posterior a la segunda generación de computadores?

RISC

337-¿En qué pareja de registros están el dato/instrucción que se leerá o escribirá en memoria, y la dirección de memoria?

MBR y MAR

338-En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:

Un incremento de las prestaciones

339-¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC es cierta?

Todas las interrupciones se pueden generar por software

340-¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso? (salva-invocante, invocado, etc)

a)
RAX, RBX, RCX, RDX

b)
CL, DX, R8d, R9

c)
RBX, RSI, RDI

d)
RSP, RBP

341-Con tres controladores de interrupciones 8259 se pueden manejar exactamente:

a)
8 niveles de prioridad

b)
16 niveles de prioridad

c)
24 niveles de prioridad

d)
Ninguna de las anteriores es cierta

342-La instrucción SKIP pone a 0 un registro o posición de memoria si se cumple una condición.

Falso

343-Si AX = FA50h y ejecutamos AND AX, 00FFh

El registro AH se pone a 0

344-El cuerpo del siguiente código C: unsigned copy(unsigned u) {return u;}; puede traducirse a ensamblador como:

movl 8(%ebp), %eax

345-¿Qué modificador (switch) de as hace falta para ensamblar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?

-32

346-En el direccionamiento inmediato, tras captarse completamente la instrucción:

se accede al operando, que es una constante contenida en la propia instrucción.

347-¿Cuál de los siguientes contenidos no está incluido en un fichero ELF ejecutable?

a)
código máquina
sección .text

b)
tabla de símbolos
incluso símbolos de depuración con -g

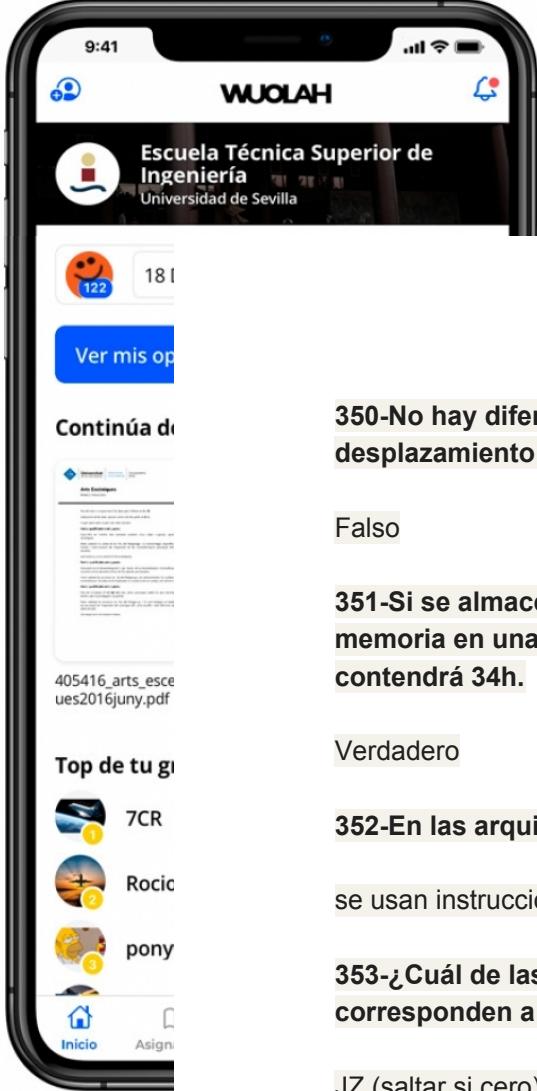
c)
variables globales
secciones .data/.bss
 d)
pila del usuario
por exclusión... y por lógica, para qué almacenarla si está vacía

348-La precaptación (cola de instrucciones) está relacionada con...

Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)

349-¿Cuántos niveles de interrupción podremos gestionar si disponemos de 7 controladores de interrupciones programables 8259?

50



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

350-No hay diferencia entre un desplazamiento lógico a la derecha y un desplazamiento aritmético a la derecha.

Falso

351-Si se almacena el número de 16 bits 1234h en la palabra (16 bits) 0 de memoria en una arquitectura "Big Endian", el byte 0 contendrá 12h y el byte 1 contendrá 34h.

Verdadero

352-En las arquitecturas RISC:

se usan instrucciones muy simples que se pueden segmentar.

353-¿Cuál de las siguientes parejas de mnemotécnicos de ensamblador IA32 corresponden a la misma instrucción máquina?

JZ (saltar si cero), JE (saltar si igual)

354-El primer parámetro de printf:

es un puntero

355-En un 8086, la ejecución de una instrucción LODSB seguida inmediatamente de una STOSB nunca modifica el contenido de ninguna posición de memoria

Falso

356-En la arquitectura Von Neumann...

el programa se encuentra residente en memoria.

357-En una microinstrucción que hace uso del registro de control residual, el contenido de éste normalmente se interpreta como señales de control.

V

358-Para direccionar una memoria de bytes en la que quepan 1G palabras de 32 bits se necesitarán:

32 bits

359-Cuando dos o más instrucciones necesitan un recurso hardware en el mismo ciclo, se trata de un riesgo:
estructural

360-La instrucción INC [3] no tiene sentido.

F

361-Cuando una CPU dispone de muchas líneas de interrupción con un dispositivo en cada una, será necesario utilizar "polling" para detectar la fuente de la interrupción.

F

362-¿Cuál de las siguientes afirmaciones sobre el direccionamiento absoluto es falsa?

El tamaño del operando direccionado queda limitado por el nº de bits del campo dirección de memoria.

363-Un programa con muchas bifurcaciones hace que no se aprovechen al máximo las prestaciones del pipeline.

V

364-Si el sistema operativo lo permite, la tabla de vectores de interrupción puede ser modificada por el usuario.

V

**365-Si el registro EAX contiene X, la sentencia en C
x &= 0x1;
se traducirá a ensamblador como:
andl \$1, %eax**

366-Al diseñar el formato de instrucción:

se suele omitir el campo que indica la siguiente instrucción (la siguiente a ejecutar es la siguiente en memoria, salvo en caso de salto).

367-Se llama "Broadcall" a la escritura simultánea en varios esclavos.

F

368-La transmisión isócrona garantiza un ancho de banda fijo sin que se produzcan interrupciones en el flujo de datos.

V

369-Sobre la E/S mapeada en memoria podemos decir que:

Usa direcciones de memoria para acceder a puertos de E/S

370-Respecto a tamaños de tipos integrales en x86 y x86-64, la excepción es que

long int pasa de 4 B a 8 B

371-En los modos de direccionamiento del tipo

Desplazamiento(Base,Indice,Factor Escala), puede usarse como

desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)

372-Una computadora puede funcionar prescindiendo de:

un acumulador

373-Los compiladores pueden generar código más fácilmente, y también código más eficiente, para conjuntos de instrucciones tipo RISC.

✓

374-Alguna de las siguientes afirmaciones sobre sistemas Linux x86-64 no es cierta

Todos los argumentos de función se pasan a través de la pila

375-Los microprocesadores RISC puede ser implementados en un tiempo más corto, y requieren menos área de silicio que los CISC.

✓

376-Dada la siguiente definición de datos:

lista: .int 0x10000000, 0x50000000,
0x10000000, 0x20000000

longlista: .int (.lista)/4

resultado: .quad 0x123456789ABCDEF

formato: .ascii "suma=%lu=%lx hex\n\0"

movl \$lista, %ebx

377-La instrucción xor \$3, %eax tiene como resultado:

Cambiar 0<->1 (complemento a 1 de) los últimos 2 bits del registro EAX

378- El direccionamiento indirecto a través de registro es más rápido que el indirecto a través de memoria.

✓

379- Una de las diferencias entre el 8086 y el 8088 es que este último no dispone de BIU ("Bus Interface Unit").

F

380- Las arquitecturas memoria-memoria tienen la ventaja de emplear instrucciones máquina muy cortas y así los accesos a memoria son mínimos.

F

381-Todas las instrucciones de salto/bifurcación realizan una operación del tipo PC = X, si bien X puede ser un dato inmediato, el contenido de un registro, o el contenido de una posición de memoria.

F

382-Un procesador emplea codificación en bloque del código de operación. Existen 123 instrucciones que tienen una longitud de 16 bits. ¿A cuántas direcciones de memoria pueden acceder como máximo si todas emplean una estructura “código de operación + dirección de memoria”?

512

383-Si una pila crece hacia direcciones de memoria decrecientes, una instrucción PUSH utiliza direccionamiento indexado con postautodecremento, siendo SP el registro índice.

F

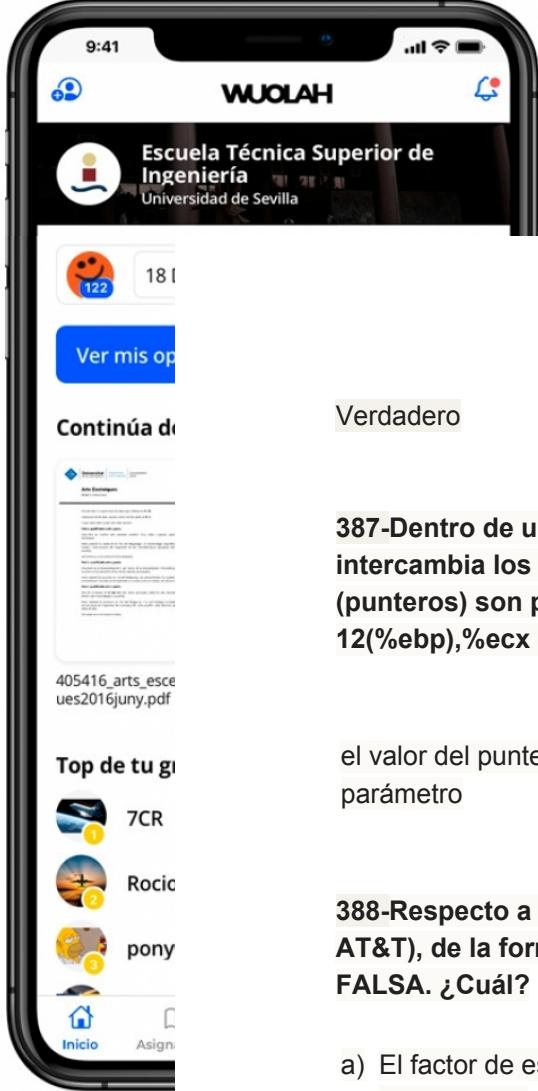
384-¿Cuántos controladores de interrupciones 8259 hacen falta como mínimo para manejar 25 líneas de interrupción?

d) 4

385-Una instrucción de "salto si igual" tiene que comprobar el valor de:

- c) el bit de cero

386-Un diseño CISC pretende disminuir el número de instrucciones a ejecutar por un programa, mientras que uno RISC pretende disminuir el número medio de ciclos por instrucción.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Continúa di

Verdadero

387-Dentro de una función declarada como void swap(int *xp, int *yp), que intercambia los valores de los dos enteros cuyas direcciones de memoria (punteros) son pasadas como parámetros a la función, la instrucción movl 12(%ebp),%ecx copia en %ecx...

el valor del puntero pasado como segundo parámetro

388-Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?

- a) El factor de escala S puede ser 1, 2, 4, 8
- b) ESP no se puede usar como registro índice
- c) EBP no se puede usar como registro base
- d) El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)

389-Para obtener una única velocidad comparativa final, el benchmark SPEC CPU combina las velocidades de ejecución de una serie de tests, respecto a un ordenador de referencia, usando la media...

- c) geométric
- a

390-Por x86-64 se entiende la misma arquitectura de repertorio (ISA) que AMD64

391-Las rutinas de servicio de interrupción son microprogramas que pueden ser escritos por el usuario.

Falso

392-En un PC, un dispositivo que pida interrupciones a través del controlador de interrupciones 8259 no podrá interrumpir por segunda vez a la CPU hasta que no mandemos el comando EOI (20h) al 8259 (puerto 20h).

VERDADERO

393-En las instrucciones aritméticas con dos operandos de un procesador con arquitectura de pila, los dos operandos...

son la cima de la pila y el elemento siguiente de la cima de la pila.

394-Si queremos almacenar la palabra de 16 bits 8965h en memoria según "big-endian", quedará almacenada a partir de la posición 1000h como en el byte 1000h se guarda 89h y en el 1001h 65h

395-¿Qué novedad se desarrolló en la tercera generación de computadores?
Los circuitos integrados

396-En el direccionamiento inmediato, la dirección efectiva se encuentra en un registro general del procesador

Falso

397-¿Cuál de las siguientes afirmaciones sobre la segmentación de cauce es cierta?

En general, un operación segmentada ("pipelined") requiere el mismo tiempo o más, desde el principio hasta el fin, que la misma operación en una implementación no segmentada

398-¿Cuál de las siguientes afirmaciones es correcta?

- a) Las instrucciones en lenguaje máquina se almacenan y tratan como cadenas de unos y ceros.
- b) Las instrucciones en lenguaje ensamblador se almacenan y tratan como cadenas de unos y ceros.
- c) El lenguaje máquina es igual para todos los computadores.
- d) El lenguaje ensamblador es igual para todos los computadores.

399-Cuando se usan interrupciones vectorizadas, el periférico puede suministrar al procesador un índice referente a una tabla de vectores de interrupción.

Verdadero

400-Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas

Falso

401-El programador de lenguaje ensamblador necesita conocer:

la arquitectura del ordenador.

402-¿Qué forma de realizar acceso directo a memoria es más rápida?

Transferencia de bloques o parada de CPU

403-En un sistema IA32 Linux, ¿cuál es el tamaño de un long?

4 bytes

404-¿Qué hace gcc -S?

Compilar .c → .s (fuente C a fuente ASM)

405-Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 6 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se aproxima a:

3,33

406-Las unidades de control microprogramadas utilizan habitualmente memorias RAM para almacenar los microprogramas.

Falso

407-Justo antes de que una instrucción máquina escriba un resultado en memoria:

en MBR está el resultado y en MAR la dirección donde se almacenará

408-En la práctica "media" se suma una lista de 32 enteros de 4 B con signo para producir una media y un resto usando la instrucción IDIV. ¿Cuál de las siguientes afirmaciones es falsa?

- a) El resto siempre tiene el mismo signo que la suma

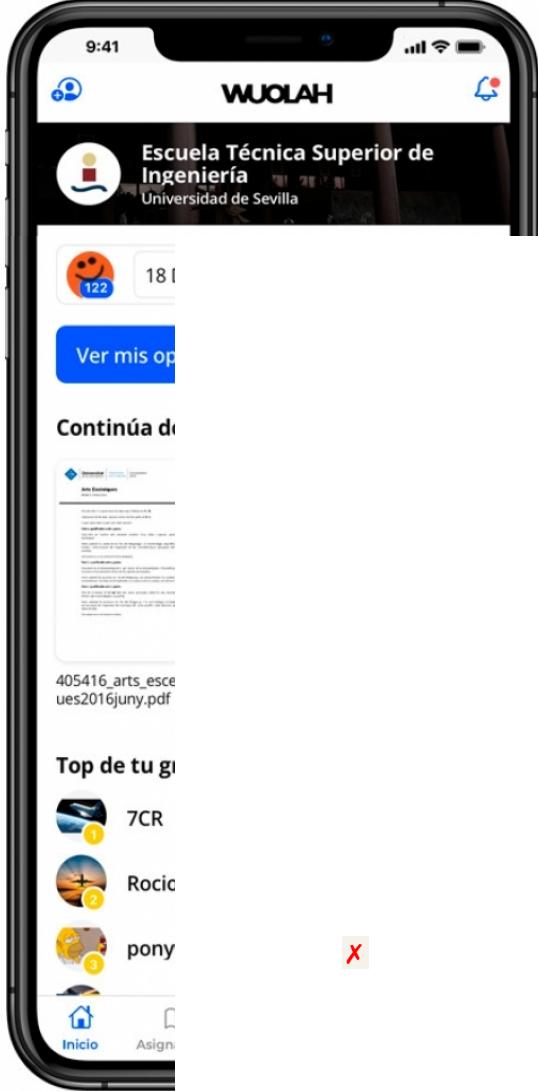
división truncada IDIV - resto del mismo signo que dividendo

En matemáticas se usa división euclídea - resto modular $0 \leq r < divisor$

https://es.wikipedia.org/wiki/Divisi%C3%B3n_eucl%C3%ADdea

•

- b) La media se redondea al entero más próximo división truncada IDIV - no se redondea, se trunca



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store GET IT ON Google Play

c) IDIV
produce el mismo resto que el operador % en lenguaje C

división truncada ambos - resto del mismo signo que dividendo

d) IDIV
produce el mismo cociente que el operador / en lenguaje C
división truncada ambos - resto del mismo signo que dividendo

409-¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?

-m elf_i386

410-¿Cuál de las siguientes afirmaciones es correcta?

Las instrucciones en lenguaje máquina se almacenan y tratan como cadenas de unos y ceros.

411-Un sistema no segmentado tarda 50 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce ("pipeline") de 5 segmentos con un ciclo de reloj de 10 ns. Cuando se procesan muchas tareas, la ganancia de velocidad que se obtiene se acerca a 5.

Verdadero

412-Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x8600 como

M[0x8600]=0x65 y M[0x8601]=0x89

413-Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.

```
struct RECORD {  
    long value2;  
    int ref_count;  
    char tag[4];  
};  
  
struct NODE {  
    double value;  
    struct RECORD record;  
    char string[8];  
};
```

También se declara una variable global "my_node" como sigue:

```
struct NODE my_node;
```

Si la dirección de my_node es 0x601040, ¿cuál es el valor de &my_node.record.tag[1]?

0x601055

414-¿Qué hace gcc -O?

Compilar con optimización, igual que -O1

415-En una bomba como las estudiadas en prácticas, del tipo...

```
0x080486e8 <main+120>: call 0x8048524 <strncmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
0x080486f6 <main+134>: ...
```

la contraseña es...

- a) el string almacenado a partir de
0x8048524
 - b) el string almacenado a partir de
0x80486f6
 - c) el string almacenado a partir de
0x8048604
 -
 - d) ninguna de las anteriores

416-Las técnicas principales de E/S son (señalar la respuesta falsa)

- a) IRQ (por
interrupciones)
 - ✓
 -
 - b) E/S cableada
(hardwired)
 - c) E/S programada
 - d) DMA (por acceso
directo)

417-Una forma usual de realizar el arbitraje distribuido consiste en una competición por la concesión del bus realizada por medio del envío por cada maestro peticionario de un número de arbitraje que lo identifica, de manera que un solo número "gane", y se le conceda el bus al maestro con ese número ganador.

VERDADERO

418-Si AX = FA50h y ejecutamos XOR AX, 00FFh

Se realiza el complemento a 1 de AL

419-La instrucción "desplazar a la derecha aritméticamente" suele ser idéntica a la "desplazar a la derecha lógicamente".

FALSO

420-Suele existir una instrucción en el repertorio de la mayoría de los microprocesadores para inhibir la petición HOLD del controlador de DMA.

FALSO

421-Respecto a los conceptos de procesamiento segmentado y superescalar, una de las siguientes afirmaciones es falsa

En cualquier procesador resulta ventajoso usar una cola de instrucciones, pero es más importante para uno segmentado (fundamental) que para uno superescalar (conveniente)

423-La CPU no guarda automáticamente el contexto del programa que ejecuta cuando le llega y atiende una petición de DMA.

Verdadero

424-La diferencia entre temporización de bus semisíncrona y asíncrona es que en la semisíncrona las transferencias ocurren en algún múltiplo de ciclo de reloj, y en la asíncrona no existe reloj del bus

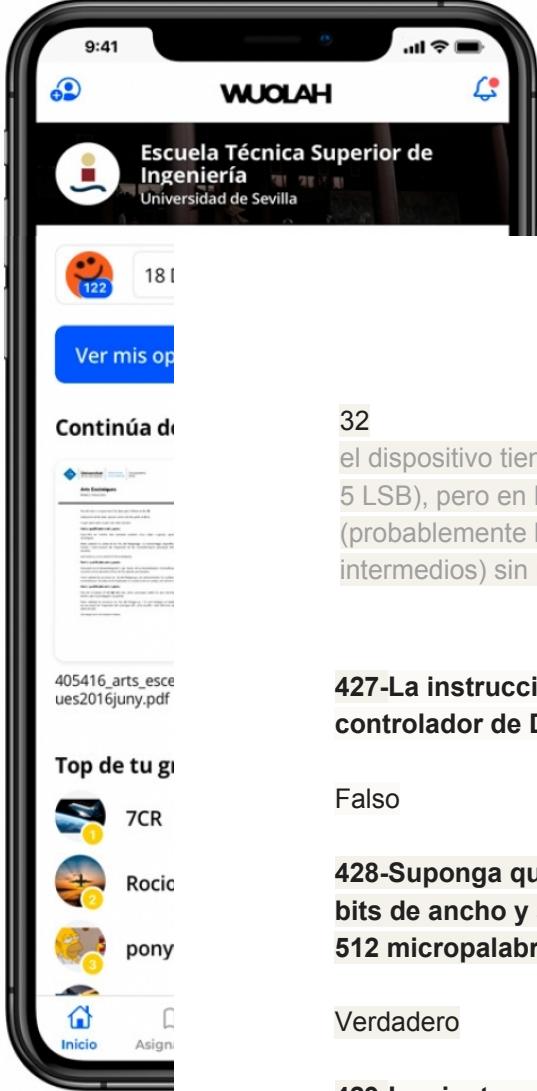
VERDADERO

425-Una unidad de control microprogramada con secuenciamiento explícito con dos direcciones por microinstrucción, tiene una memoria de control con 35 bits de longitud de palabra. Si las microinstrucciones emplean 15 bits en total para los campos de control y de tipo y condición de salto, el número máximo de palabras de la memoria de control de esta unidad de control microprogramada es de:

2^{10}

426-Si en un bus de direcciones de 32 bits se decodifica parcialmente la dirección de un dispositivo de 32 posiciones usando 22 bits, ¿cuántas veces aparecerá repetido en el mapa de memoria?

32



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

32

el dispositivo tiene 32 puertos, usa 5 bits de direccionamiento (probablemente los 5 LSB), pero en lugar de decodificarse con 27, se usan sólo 22 bits (probablemente los 22 MSB) de manera que quedan 5 bits (probablemente intermedios) sin usar, que pueden tomar 32 combinaciones posibles

427-La instrucción DISABLE (CLI en el 8086) inhibe la petición HOLD del controlador de DMA.

Falso

428-Suponga que la micropalabra de una máquina microprogramada tiene 8 bits de ancho y se usan 256 micropalabras diferentes en un microprograma de 512 micropalabras. No se ahorran bits usando nanoprogramación.

Verdadero

429-Las instrucciones máquina más usadas según el análisis dinámico son las de bifurcación.

Falso

430-El ajuste de marco de pila que gcc (Linux/IA-32) prepara para todas las funciones consiste en las instrucciones

```
pushl %ebp  
movl %esp,  
%ebp
```

431-Algunas de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente son:

El diseño de la CPU es más sencillo.

432-En la práctica “media” se pide sumar una lista de 32 enteros *con* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista causaría overflow con 32bits?

0xfbff
ffff

433-La transferencia de datos en un computador y los dispositivos de E/S puede manejarse de diversos modos. Uno de los siguientes es falso; indíquelo:

Manejo de todas las líneas del bus de control, paralizando la CPU

434-Las señales de carga/incremento/desplazamiento de registros, las de selección de entradas de multiplexores, y las de selección de función de la ALU son entradas a la unidad de control

FALSO

435-¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila: push #4; push #7; add; push #10; sub?

1

436-Un controlador de DMA de un sistema de que emplee buses separados avanzados suele ser programado con la siguiente información relativa a una operación de E/S

tipo de operación, tamaño de bloque a transferir, dirección inicial de memoria

437-¿Por qué se impusieron las arquitecturas de registros de propósito general a las arquitecturas basadas en pila?

Porque las basadas en registros son capaces de lograr un mejor rendimiento cuando se asignan variables a registros

438-¿Cuál de los siguientes microprocesadores no es de 64 bits?

Pentium III

439-¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC (modo real) es cierta?

a) Existen

1024
vectores de
interrupción



- b) Cada vector de interrupción es una doble palabra de 32 bits formada en primer lugar (dirección menor) por el segmento y seguida por el desplazamiento (dirección mayor) de cada rutina de servicio de interrupción
- c) No todas las interrupciones se pueden generar por software
- d) Ninguna de las anteriores afirmaciones es cierta

440-Un procesador emplea codificación en bloque del código de operación. Existen 120 instrucciones que tienen una longitud de 16 bits. ¿A cuántas direcciones de memoria pueden acceder como máximo si todas emplean una estructura "código de operación + dirección de memoria"?

512

441-El direccionamiento a registro es similar al directo. La diferencia es que el campo de direcciones referencia un registro en lugar de una dirección de memoria

VERDADERO

442-La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:

```
int popcount3(unsigned* array,
              int len){
    int i, res = 0;
    unsigned x;
    for( i=0; i<len; i++ ) {
        x = array[i];
        asm("ini3: \n"
            "shr %[x] \n"
            "adc $0, %[r] \n"
            "add $0, %[x] \n"

```

```
"jne ini3 \n"
: [r] "+r" (res)
: [x] "r" (x );
}
return res;
}
```

Esta función produce siempre el resultado correcto, a pesar de que una instrucción máquina en la sección asm() es distinta a la que se esperaba después de haber estudiado pcount_r en teoría. La instrucción distinta también se podría haber cambiado por test %[x], %[x]

443-¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop:

```
push #1
push #2
push #3
pop a
push #4
pop a
pop a
```

1

444-En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x0400 0000, ¿en qué se diferencian los resultados de ambos programas?

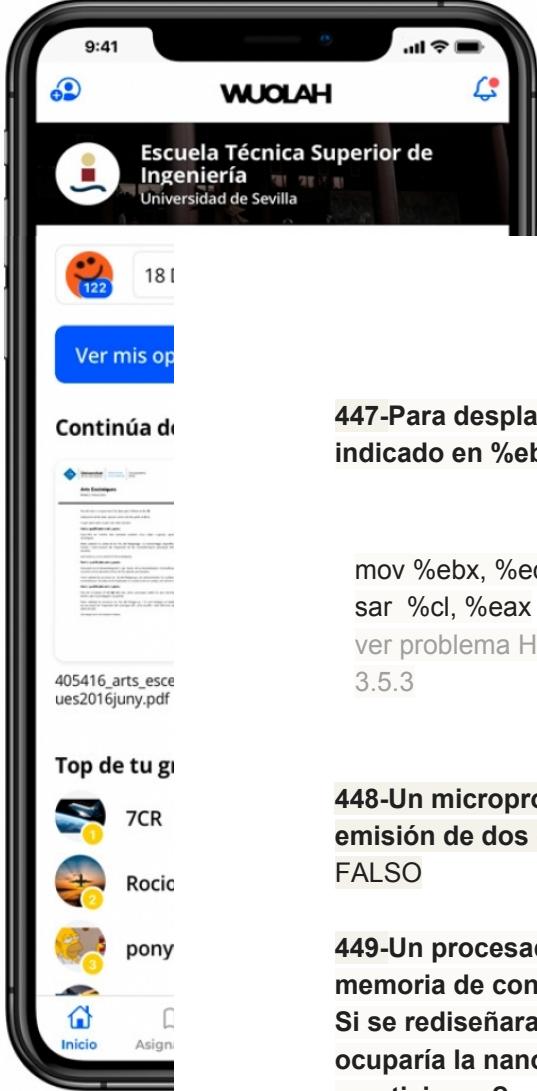
no se diferencian

445-El campo tipo de secuenciamiento indica al generador de direcciones de una unidad de control microprogramada el mecanismo de cómputo de la EA del operando de la instrucción.

FALSO

446-La conexión de un 8086 a un sistema de memoria y E/S requiere algún circuito externo más en modo máximo que en modo mínimo.

Verdadero



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

447-Para desplazar %eax a la derecha un número variable de posiciones <= 32, indicado en %ebx, se puede hacer

mov %ebx, %ecx
sar %cl, %eax
ver problema Hallaron 3.8 y sección
3.5.3

448-Un microprocesador es superencauzado ("superpipelined") si permite la emisión de dos o más instrucciones en un mismo ciclo de reloj
FALSO

449-Un procesador con una unidad de control microprogramada tiene una memoria de control de 300 palabras de 100 bits, de las que 200 son diferentes. Si se rediseñara como unidad de control nanoprogramada, ¿qué tamaño ocuparía la nanomemoria que contiene las microinstrucciones completas sin repeticiones?

20000 bits
200 uinstr. x 100 bits

450-Una máquina que almacene el número 4321h a partir de la dirección 0 como 21h, 43h utiliza el sistema little-endian.
VERDADERO

451-En las arquitecturas de registros de propósito general suele ser menor el tráfico entre procesador y memoria que en las de acumulador.

Verdadero

452-En la codificación por extensión de campo, el campo de código de operación aumenta su tamaño a medida que disminuye el número o la longitud de campos de dirección de operandos.

VERDADERO

453-

Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para

escribir el resultado en el banco de registros, ¿cuál es la frecuencia de reloj máxima del procesador?

66,67 MHz

454-El especificador de operando de una instrucción, cuando existe, es siempre una dirección de memoria o de entrada/salida.

F

455-¿Cuál de las siguientes instrucciones de IA32 (en sintaxis Intel) no es una instrucción de transferencia?

cmp eax,15h

456-El análisis dinámico de la frecuencia de utilización de instrucciones se realiza mediante el estudio de un gran número de listados de programas.

F

457-¿Cuál de las siguientes instrucciones convierte %eax = 5 * %eax?

- 1) mov (%eax, %eax, 4), %eax
- 2) lea (%eax, %eax, 4), %eax

Sólo 2

458-La codificación Huffman es la más utilizada debido a que consigue resultados óptimos y su proceso de decodificación es sencillo.

F

459-Un computador que almacene el número 2143h como 43h en el byte de la dirección 0 y 21h en el byte de la dirección 1, utiliza el sistema big-endian.

F

460-¿Cuál de las siguientes afirmaciones es incorrecta?

- a) En lenguajes de alto nivel no se tiene acceso a detalles del procesador como las tablas de páginas, el paso de modo usuario a modo supervisor, el bit de paridad, etc.
- b) El lenguaje de alto nivel es más portable que el lenguaje máquina.

- c) En lenguaje ensamblador cada instrucción se corresponde con una instrucción máquina.
- d) En lenguaje ensamblador las instrucciones se escriben en binario.

461-El direccionamiento relativo a contador de programa favorece la implementación de código reubicable.

V

462-Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?

$$k + n \cdot (2^m - 1)$$

463-¿De qué depende el tamaño del contador de programa?

Del número de direcciones de memoria.

464-En X86-64, el registro contador de programa se denomina:

RIP

465-En una CPU de 32 bits con memoria de bytes, el problema es que...

Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU

466-En el direccionamiento inmediato:

el tamaño (rango de valores) de la constante está limitado.

467-Un procesador emplea codificación en bloque del código de operación. Existen 130 instrucciones que tienen una longitud de 16 bits. ¿A cuántas direcciones de memoria pueden acceder como máximo si todas emplean una estructura "código de operación + dirección de memoria"?

256

468-Se puede programar un controlador de interrupciones 8259 de manera que atienda equitativamente a 8 dispositivos de igual prioridad (cada vez que se atiende a un dispositivo, éste pasa automáticamente a tener la prioridad más baja)

VERDADERO

469-¿Cuál es el resultado de evaluar la expresión 0b1110 ^ 0b1010 en lenguaje C?

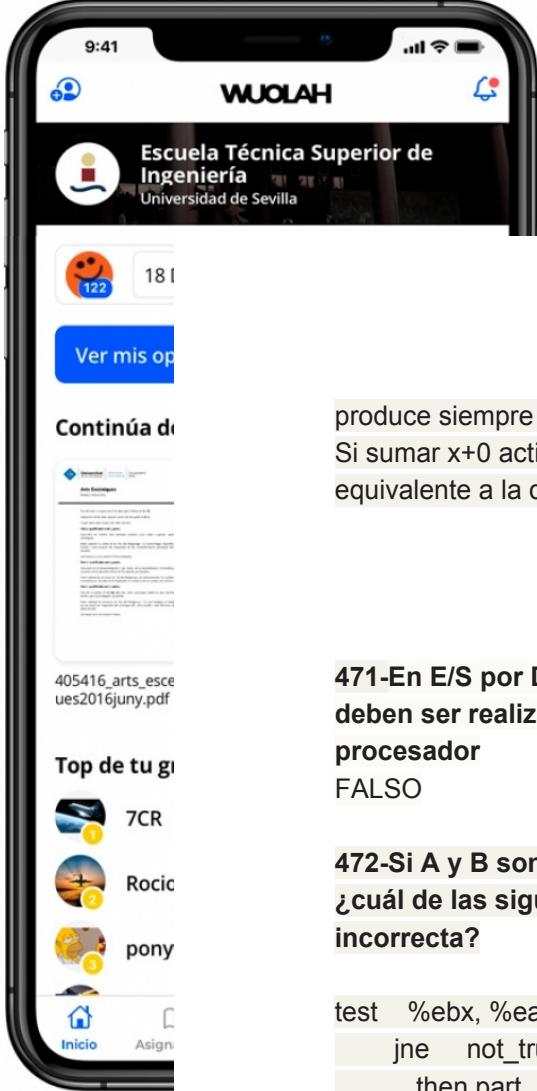
0b0100 (hace el AND y pone 1 cuando no es AND, 1 y 1 = 0, 0 y 1 = 1)

470-La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de `popcount3`:

```
int popcount3(unsigned* array,
              int len){
    int i, res = 0;
    unsigned x;
    for( i = 0; i < len; i++ ) {
        x = array[i];
        asm("ini3: \n"
            "shr %[x] \n"
            "adc $0, %[r] \n"
            "add $0, %[x] \n"
            "jne ini3 \n"
            : [r] "+r" (res)
            : [x] "r" (x) );
    }
    return res;
}
```

Esta función sólo tiene una diferencia con la versión "oficial" recomendada en clase. En concreto, una instrucción máquina en la sección `asm()` es distinta.

Esta función `popcount3`:



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

produce siempre el resultado correcto

Si sumar $x+0$ activa ZF sólo puede ser porque ya era $x==0$, así que la lógica es equivalente a la deseada.

471-En E/S por DMA, las operaciones elementales de transferencia de datos deben ser realizadas por medio de la ejecución de instrucciones del procesador

FALSO

472-Si A y B son dos enteros almacenados respectivamente en %eax y %ebx, ¿cuál de las siguientes implementaciones de if (!A && !B) {...then part...} es incorrecta?

```
test %ebx, %eax
    jne not_true
    ...then part...
not_true:
    ...
test hace AND, no OR
equivale a if (A&&B) {... then_part ...}
```

473-Si el contenido de r4 es 0x13000, ¿a qué dirección se hace referencia con la instrucción LD -0x80(r4)?

(Nota: 0x indica notación hexadecimal en C y C++)

0x12F80

474-¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar)

a)
movw %dx, (%eax)

b)
movzbl %dl, %eax

c)

movswl (%eax), %edx

•d)

pushb \$0xFF

475-El "handshaking" se utiliza en transferencias de datos "síncronas" entre CPU y periféricos.

Falso

476-El primer microprocesador de 32 bits de la familia x86 fue 80386

477-Si el acceso directo a memoria se realiza mediante robo de ciclo:
es posible que la ejecución de una instrucción máquina sea temporalmente detenida

478-Un cauce ("pipeline") de instrucciones inicialmente vacío y con 3 etapas tardará siempre 5 ciclos de reloj en ejecutar 3 instrucciones si cada una de ellas utiliza las 3 etapas.

Falso

479-Para direccionar una memoria de 2 G palabras de 32 bits cada una, que se direcciona byte a byte, se necesitarán:

33 bits como mínimo

480-En los RISC la sección de control del procesador debe ser microprogramada en lugar de cableada debido al uso de un pequeño número de operaciones simples.

481-¿En qué generación, dentro de la historia de los computadores digitales, aparecen los sistemas operativos multiusuario?
tercera

482-¿Qué dice la ley de Moore?

Que el número de transistores de un chip se duplica cada 18 meses.

483-En el acceso directo a memoria la CPU pone en el bus de direcciones del sistema las direcciones de memoria correspondientes a cada dato que se transfiere por DMA.

FALSO

484-¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?

Una memoria.

485-¿Con cuál de los siguientes dispositivos tendría sentido utilizar E/S programada sin consulta de estado?

Salida a un display de 7 segmentos

486-Después de ejecutar el siguiente código, ¿qué variables serán igual a 0? (Suponer ints de 32bits y longs de 64bits)

```
unsigned int a = 0xffffffff;
unsigned int b = 1;
unsigned int c = a + b;
unsigned long d = a + b;
unsigned long e = (unsigned long)a + b;
```

c y d

En el problema 3.4 sólo se explica que una extensión de tamaño se hace según el fuente sea signed (extensión sgn) o unsigned (ext. con ceros), pero no se explica la sección 2.2.6 (y 2.2.5) en donde se aclara que las operaciones que impliquen a algún unsigned se hacen en unsigned. Evitar esta pregunta en el futuro.

487-Un microprocesador es superencauzado ("superpipelined") si permite la emisión de dos o más instrucciones en un mismo ciclo de reloj.

Falso

488-Una arquitectura de registros de propósito general puede tener instrucciones máquina de la ALU con un único operando explícito.

Verdadero

489-¿Qué tipo de instrucciones se emplean más en una arquitectura de acumulador?

de transferencia de datos con memoria

490-¿Qué modificador (switch) de gcc hace falta para compilar .s → .o sin llamar al enlazador?

gcc -c

491-Si queremos almacenar la palabra de 16 bits 9660h en memoria según "little-endian", quedará almacenada a partir de la posición 1000h como:

en el byte 1000h se guarda 60h y en el 1001h se guarda 96h

492-En una resta de dos números en complemento a dos, se produce desbordamiento cuando...

a)

los dos operandos son negativos y el resultado es positivo.

b)

los dos operandos son positivos y el resultado es negativo.

x

c)

Las dos respuestas a y b son correctas.

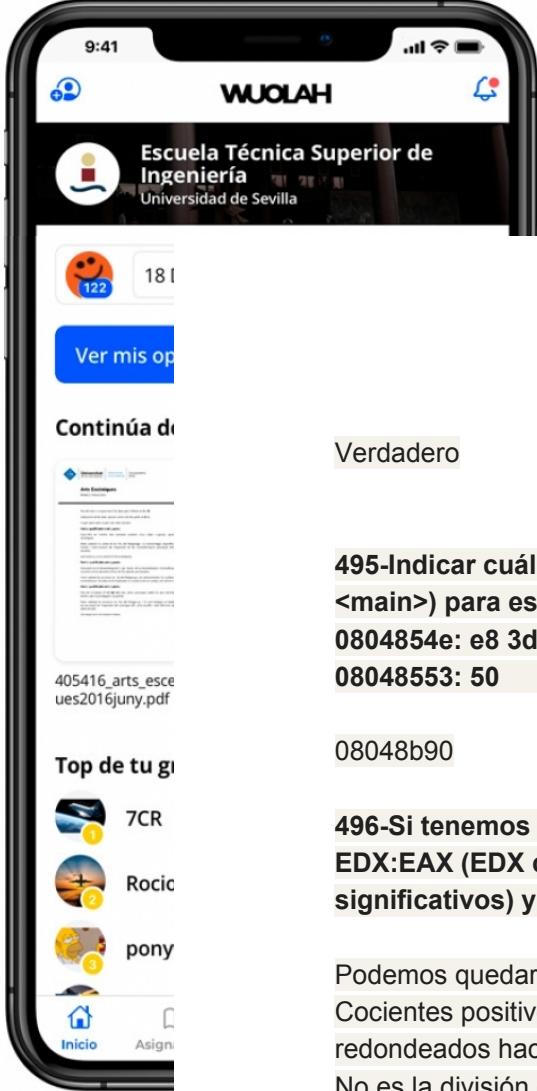
d)

Ninguna de las anteriores es correcta.

493-Si el registro %edx contiene la variable definida con la sentencia en C “int n”, y queremos dividirla por 2, usaremos la instrucción:

sarl %edx,1

494-Una de las funciones de una interfaz de E/S es realizar la conversión de cualquier formato que pueda ser necesaria para transferir datos entre el bus de E/S y el dispositivo de E/S (por ejemplo, de paralelo a serie).



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Verdadero

495-Indicar cuál es la dirección de salto (en qué dirección empieza la subrutina <main>) para esta instrucción call

0804854e: e8 3d 06 00 00 call <main>

08048553: 50 pushl %eax

08048b90

496-Si tenemos un número n, de 64 bits, almacenado en la pareja de registros EDX:EAX (EDX contiene los 32 bits más significativos y EAX los 32 bits menos significativos) y queremos realizar la división n / (2^32) entonces:

Podemos quedarnos con EDX tanto si n es un número con signo como sin signo. Cocientes positivos o sin signo redondeados hacia cero, cocientes negativos redondeados hacia más negativo: es decir, redondeado siempre hacia inferior. No es la división entera estándar de C (que redondea hacia cero) pero el enunciado tampoco lo exige.

497-¿Cuántos puertos de E/S permite manejar la interfaz de periféricos programable 8255?

- c) 4 puertos de 32 bits

498-Con respecto a los buses de la placa base:

El bus ISA es un bus

- a) local.

X

•

- b) El bus EISA es un bus local.
- c) El bus MCA es un bus local.
- d) Ninguna de las anteriores es cierta.

499-Una función C llamada get_el() genera el siguiente código ensamblador. Se puede adivinar que:

```
movl 8(%ebp), %eax
leal (%eax,%eax,4), %eax
addl 12(%ebp), %eax
movl var(%eax,4), %eax
```

var es un array bidimensional de enteros, con cinco columnas

se puede adivinar incluso el código:

```
int get_el(int fila, int columna){
    return var[fila][columna];
}
```

porque se calcula %eax=fila*5+columna para indexar en var(%eax,4)

500-La práctica “parity” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity4:

```
int parity4(unsigned* array, int len){
    int val,i,res=0;
    unsigned x;
    for (i=0; i<len; i++){
        x=array[i];
        val=0;
        asm("\n"
            "ini3:      \n\t"
            "xor %[x],%[v] \n\t"
            "shr %[x]     \n\t"
            "test %[x], %[x]\n\t"
            "jne ini3   \n\t"
            "[v]" +r" (val)
            "[x]" "r" (x)
            );
        val = val & 0x1;
        res+=val;
    }
    return res;
}
```

La sentencia asm() del listado anterior tiene las siguientes restricciones

dos entradas y una salida

- c) [v] cuenta como salida y entrada, [x] como
entrada

**501-¿Cuántas patillas de dirección tiene una memoria DRAM de 1G palabra,
siendo la longitud de palabra de 16 bits?**

15

**502-Una memoria que está estructurada en palabras de 8 bits tiene una
capacidad de 64 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?**

13

503-Las celdas de memoria estática...

mantienen la información almacenada por tiempo indefinido mientras se
mantenga la alimentación

504-Toda celda de memoria DRAM tiene al menos cuatro transistores.

FALSO

**505-Una organización de interrupciones de niveles múltiples con prioridad
significa que, durante la ejecución de una rutina de servicio de interrupción, se
aceptarán solicitudes de interrupción de algunos dispositivos, pero no de
otros, según sea su prioridad**

**506-¿Cuál de los siguientes grupos de señales son entradas a la unidad de
control?**

- a) Las señales de
carga/incremento/desplazamiento
de registros
- b) Las señales de selección de entradas
de multiplexores del datapath
-
- c) Los bits del registro de indicadores
(flags)
- d) Los bits de las opciones b y c

X

507-Para direccionar una memoria de bytes en la que quepan 2G palabras de 32 bits se necesitarán:

33 bits como mínimo

508-Los riesgos de datos consisten en que...

una instrucción necesita un dato calculado por otra anterior

509-Las unidades de control cableadas memorizan los pasos de ejecución de una instrucción máquina en una memoria de control, y las microprogramadas en una PLA.

Falso

510-La instrucción IA32 test sirve para...

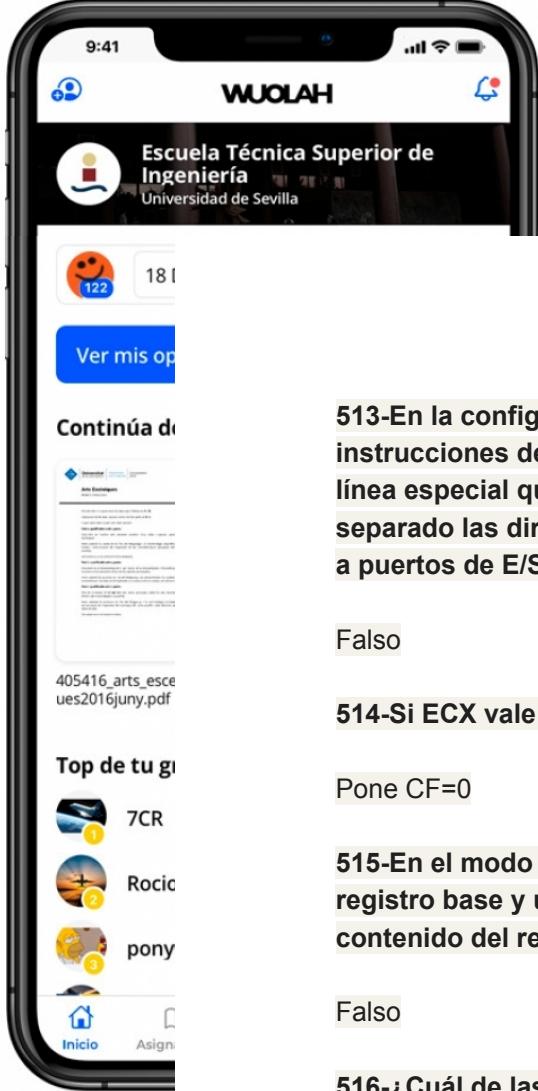
Realizar la operación and lógico bit-a-bit (a&b) pero no guardar el resultado, sino simplemente ajustar los flags

511-Respecto a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida

la selección lineal permitiría escribir un mismo dato a varios puertos E/S

512-¿Cuál sería el "equivalente x86-64" del "pseudo-código C" `rcx = ((int*)rax)[rcx]`?

`mov (%rax,%rcx,4),%rcx`



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

513-En la configuración de E/S mapeada en memoria, la CPU tiene instrucciones de E/S, y cuando se ejecuta una de ellas, la CPU habilita alguna línea especial que sirve para que la circuitería externa decodifique por separado las direcciones correspondientes a memoria y las correspondientes a puertos de E/S.

Falso

514-Si ECX vale 0, la instrucción adc \$0,%ecx

Pone CF=0

515-En el modo de direccionamiento relativo a registro base se especifica un registro base y una dirección de memoria cuyo contenido se suma al contenido del registro base para obtener la dirección efectiva.

Falso

516-¿Cuál de las siguientes afirmaciones sobre el benchmark SPEC CPU es falsa?

El resultado final es la media aritmética de las (12 ó 17) velocidades, bien sea de enteros ó de punto flotante (SPECint2006 ó SPECfp2006)

517-¿Qué valor contendrá %edx tras ejecutar las siguientes instrucciones?

```
xor %eax, %eax  
sub $1, %eax  
cltd  
idiv %eax
```

0

518-¿Cuál de las siguientes no es una sección de un fichero ELF?

- a)
.text
para el código
- b)
.static

inventada

c)

.data

para datos inicializados como en la Práctica "media"

x d)

.bss

para datos sin inicialización, mencionada en P2 Apéndice 2 Tabla 9

519-La evolución de la arquitectura de computadores a lo largo de la historia de éstos se ha producido de forma desligada a la evolución de la tecnología de fabricación de dispositivos electrónicos.

Falso

520-El incremento de velocidad de los RISC se consigue a costa de un aumento del área de chip dedicado a la unidad de control.

Falso

521-El objetivo del control residual es aumentar la velocidad de ejecución de microinstrucciones, aunque esto tiene el inconveniente de aumentar el tamaño del microprograma.

FALSO

522-¿De qué depende el tamaño del contador de programa?

Del número de direcciones de memoria.

523-En un sistema con un único bus...

a) sólo un dispositivo puede escribir en un instante dado en el bus

- b) se utilizan las mismas líneas de control para conectar todos los dispositivos
- c) el procesador y los periféricos pueden funcionar a diferentes velocidades si el funcionamiento del bus es asíncrono



•

- d) Todas las respuestas anteriores son ciertas

524-Escribiendo en el registro de máscara de interrupciones es posible inhabilitar todas las interrupciones enmascarables.

Verdadero

525-¿Cuál de las siguientes no es una característica de los computadores RISC?

La decodificación de las instrucciones

- a) debe ser simple: un computador RISC debería emplear un único formato de instrucción
-
- b) Para acelerar el computador RISC se emplean técnicas de pipelining.
- c) Las funciones que realizan los computadores RISC deben ser lo más complejas y potentes que sea posible.
- d) Un computador RISC no debe emplear microprogramación.

526-Los procesadores RISC no tiene instrucciones para sumar un registro con una posición de memoria.

Verdadero

527-En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor que se indica a continuación, ¿en qué caso ambos programas producen el mismo resultado?

0x1111 1111

resultado 0x0000 0002 2222 2220

porque es positivo incluso en complemento a 2

todos los demás valores se interpretan como negativos, lo primero que hace la suma con signo es extenderlos a 64bit de manera que se activan los 32 bits superiores... resultado radicalmente distinto

528-El entrelazado de orden superior se utiliza más que el de orden inferior en procesadores vectoriales debido a que su expansión es más fácil.

FALSO

529-La expresión $d(t+k) = d(t)$, donde $d(x)$ es la dirección de memoria referenciada en el instante de tiempo x y k es un pequeño incremento de tiempo, corresponde al principio de localidad espacial.

FALSO

530-La penalización por una falta de página suele ser de unas pocas decenas de ciclos de reloj de la CPU.

Compilar sin optimización

531-De las siguientes parejas de instrucciones, ¿cuál utiliza únicamente direccionamiento implícito?

- a) mul,
div
- b) push,
call
- c) jnz, cmp
- d) sti, popf

532-Suponga un ordenador que usa un tamaño de palabra de 32 bits y un espacio de direccionamiento de 20 bits. El valor hexadecimal de la dirección más alta es FFFFFFFFh.

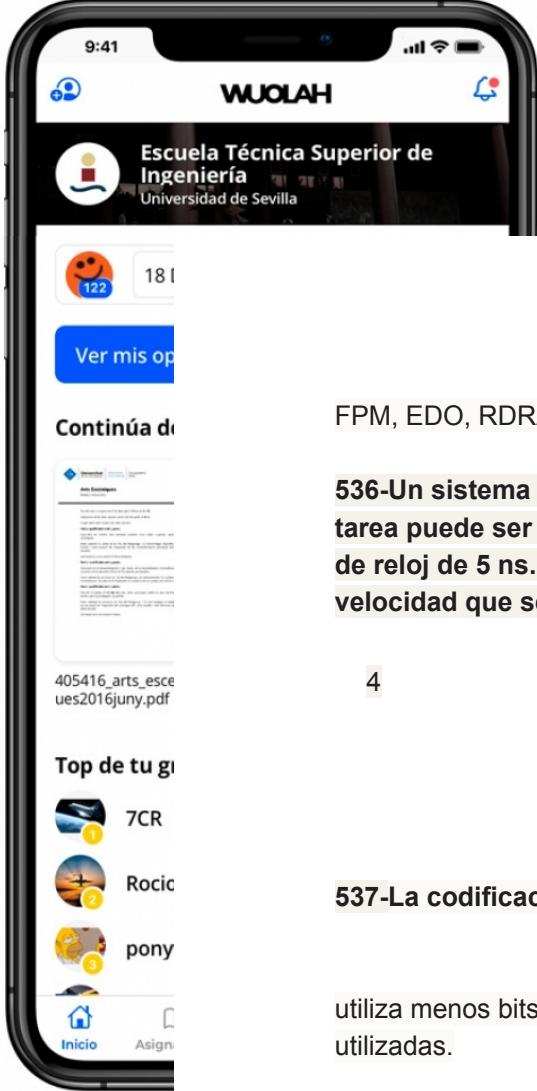
FALSO

533-¿Qué hace gcc -O0?

534-El 8255 es un interfase paralelo de periféricos que suele utilizarse para E/S programada, pero también puede generar interrupciones.

VERDADERO

535-¿Cuál de las siguientes secuencias de tipos de memoria está ordenada de menores a mayores prestaciones?



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

FPM, EDO, RDRAM

536-Un sistema no segmentado tarda 20 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se acerca a:

4

537-La codificación Huffman...

utiliza menos bits para las instrucciones más frecuentes y más bits para las menos utilizadas.

538-La penalización por una falta de página suele ser de unas pocas decenas de ciclos de reloj de la CPU.

FALSO

539-En un procesador capaz de direccionar la memoria a nivel de bytes, con una longitud de palabra y tamaño del bus de datos de 32 bits, ¿cuántos accesos a memoria hacen falta para llevar de la memoria al procesador un objeto de tamaño doble palabra situado a partir del byte de dirección 4 (inclusive)?

2

540-El primer microprocesador de 32 bits de la familia x86 fue el:

80386

541-En los procesadores RISC todas las instrucciones son del tipo registro-registro, exceptuando las de movimiento de datos entre registros.

Falso

542-El acceso a memoria en modo página no se utiliza en segmentación no paginada.

Falso

543-¿Qué técnica de E/S se dice controlada por hardware?

Acceso directo a memoria

544-GCC/Linux IA32 resuelve el ajuste de marco de pila mediante las instrucciones:

pushl %ebp; movl %esp, %ebp

545-En una memoria virtual con segmentos paginados, cada entrada de la tabla de segmentos contiene un puntero a la primera dirección física del segmento correspondiente.

Falso

546-Respecto a los términos microinstrucción y microcódigo:

Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoria

tr.42 define literalmente ucód. como conjunto de microprogs. que a su vez son el conjunto de uinstr. de una instr.

547-Sólo los microprocesadores RISC utilizan segmentación en la ejecución de sus instrucciones.

FALSO

548-SP y PC no son registros de uso general (GPR).

VERDADERO

549-La propiedad de localidad espacial se puede enunciar como: "la información que se usará en un futuro próximo es la misma que se está usando actualmente".

Falso

550-Un procesador de 1GHz tarda 4ns en realizar 4 instrucciones sin realizar segmentación de cauce. ¿Cuanto tardaría en realizar 9 instrucciones un procesador con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones?

3 ns

551-Cuando se usan interrupciones vectorizadas, el periférico siempre suministra a la CPU la dirección de memoria (completa o incompleta) donde comienza la rutina de servicio de interrupción.

FALSO

552-RAS significa impulso de acceso a filas o selección de acceso a filas.

VERDADERO

553-¿Cuál de las siguientes afirmaciones es falsa?

a)

el bus de control transporta señales de estado

x

b)

el bus de direcciones es unidireccional

c)

el bus de datos es bidireccional

d)

la anchura del bus de datos es de 16 bits

554-No existen instrucciones máquina que únicamente contengan campo de operación.

FALSO

555-La eficiencia de un sistema que emplea memoria caché (definida como la razón entre el tiempo de acceso a caché y el tiempo medio de acceso al sistema caché-M.P.) vale 0 cuando la razón de aciertos de la caché vale también 0.

VERDADERO

556-En una arquitectura RISC típica:

se usan muchas instrucciones de las disponibles en el conjunto de instrucciones.

557-Si un byte puede agrupar píxeles consecutivos, un modo de vídeo de 512 x 256 píxeles y 16 colores por píxel ocupa una memoria de:

64 KB

558-La instrucción movl %esp,%ebp

Copia el contenido del registro ESP en el registro EBP.

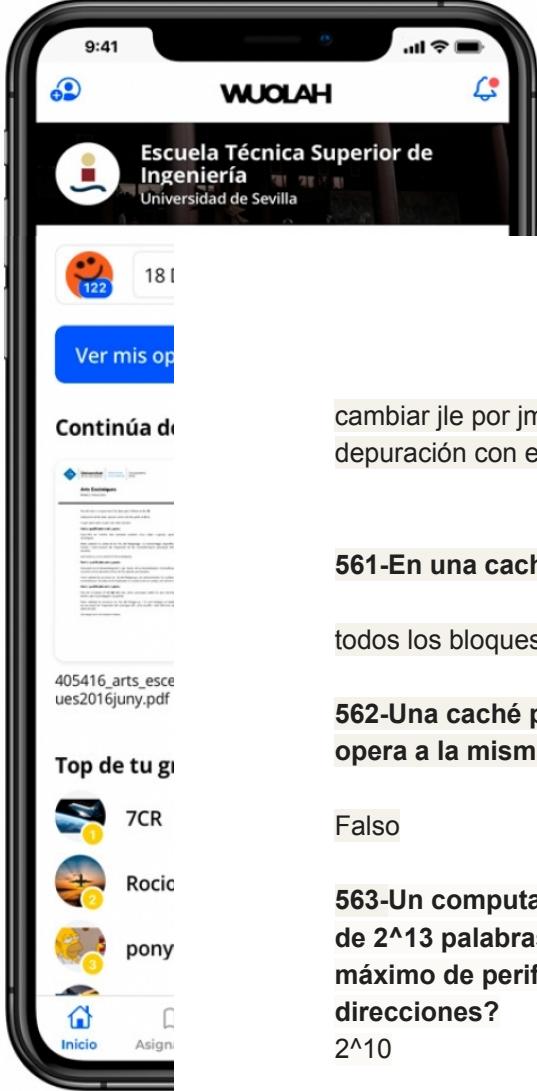
559-Es más caro emplear buses dedicados que no dedicados, pero ello permite establecer más comunicaciones simultáneamente, aumentando las prestaciones de un computador.

VERDADERO

560-En una bomba como las estudiadas en prácticas, del tipo...

```
0x08048705 <main+149>: call 0x80484c4    <gettimeofday>
...
0x08048718 <main+168>: cmp $0x5,%eax
0x0804871b <main+171>: jle 0x8048722 <main+178>
0x0804871d <main+173>: call 0x8048604 <boom>
0x08048722 <main+178>: ...
```

ejecutada paso a paso con el depurador ddd, interesaría...



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

cambiar jle por jmp usando ddd o un editor hex, salvar el programa y reiniciar la depuración con el nuevo ejecutable

561-En una caché asociativa por conjuntos, la vía i está constituida por:

todos los bloques i-ésimos de cada conjunto

562-Una caché puede incrementar las prestaciones de la memoria incluso si opera a la misma velocidad que la memoria principal.

Falso

563-Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de 2^{13} palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?

2^{10}

564-Para un tamaño de línea fijo, llega un momento en que la tasa de faltas aumenta al aumentar el tamaño de la memoria caché.

Falso

565-En las transferencias asíncronas, es común acompañar a cada dato transferido de una señal de control que indica la presencia del dato en el bus.
VERDADERO

566-¿En qué tipo de traducción de memoria virtual se utilizan los campos base y límite?

Segmentación

567-En el contexto de una llamada a función cdecl: si los contenidos de ESP y EBP son, respectivamente, 0x0008d040 y 0x00000000 antes de ejecutar una instrucción call, tras ejecutar el ret correspondiente, los contenidos serán: 0x0008d040 y 0x00000000

568-La directiva de ensamblador "DW 100 DUP(0)" reserva 100 palabras inicializadas a cero.

Verdadero

569-En un sistema con dos buses separados, uno para el subsistema de memoria y otro para la E/S...

el bus que une la memoria y el procesador suele funcionar a la velocidad de la memoria

570-El marco de pila en x86-64 Linux...

se crea para funciones en las que GCC no puede evitar que RSP baje más, como por ejemplo: que haya que calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función

571-El objetivo de un diseño CISC es...

disminuir el número de instrucciones a ejecutar por un programa.

572-¿Cuál de las siguientes afirmaciones es falsa?

Las memorias DRAM

- a) presentan generalmente una capacidad de almacenamiento mucho mayor que las SRAM.
- b) La lectura de un bit de la matriz de almacenamiento de una memoria DRAM proporciona una señal mucho más débil que la suministrada por los inversores de una celda de memoria SRAM.
- c) Las memorias DRAM son en general mucho más rápidas que las SRAM
- d) Una celda DRAM sólo necesita un transistor y un condensador.

573-¿Cómo actúa el indicador Z del registro de indicadores de estado?

Se pone a 1 cuando el resultado de una operación es 0

574-Los buses de placa madre ("board level") son relativamente rápidos y no requieren terminadores.

VERDADERO

575-Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,  
        0x10000000, 0x20000000  
longlista: .int (. -lista)/4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\0"
```

y suponiendo que hemos llamado a una función suma que devuelve un número de 64 bits en la pareja EDX:EAX, las instrucciones que copian ese número en resultado son:

```
movl %eax, resultado  
movl %edx, resultado+4  
little-endian => primero el menos significativo
```

576-Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta 16 pastillas

577-En una jerarquía de memoria, a medida que nos alejamos del procesador...

El tamaño de la unidad de transferencia entre dos niveles aumenta.

578-¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?

- En lenguaje C, al llamar a una
- a) subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
 - b) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable

-
- c) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop
- X
- d) Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero

579-Un computador emplea un sistema de memoria principal de 128 palabras y una memoria caché de 32 palabras. La organización de la memoria caché es totalmente asociativa y el tamaño de bloque es de 8 palabras. Se emplea el algoritmo de reemplazo FIFO. Si inicialmente la memoria caché está totalmente vacía, calcule el número de fallos cuando se lee la secuencia de direcciones de la memoria principal: 0000100, 1000001, 0000101, 0010011, 0100010, 1000100, 0000111.

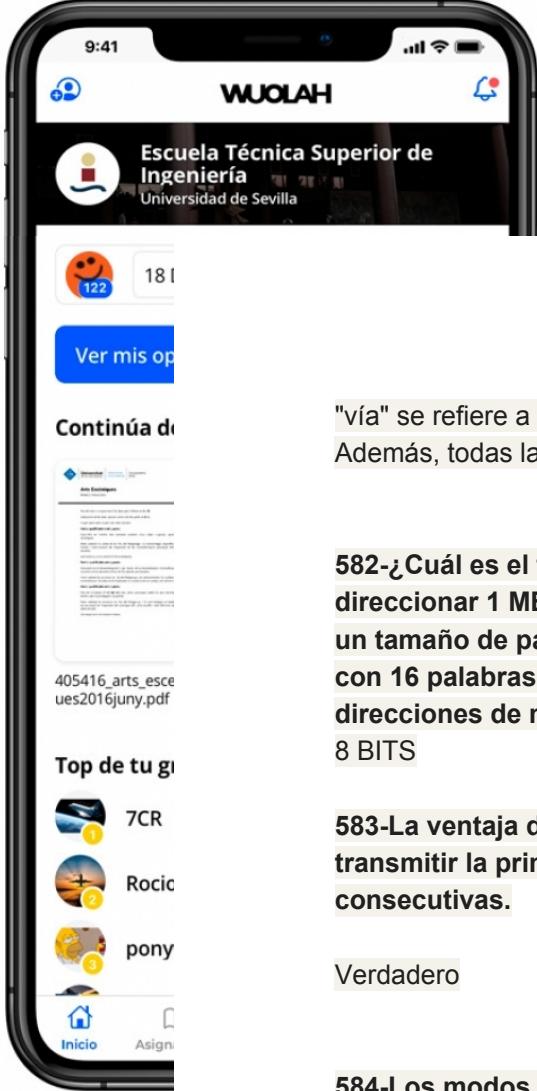
- 6
- a) fallos
- b) 3
fallos
- ✓ • c) 4
fallos
- d) 5
fallos

580-Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?

Los registros base e índice (Rb y Ri) pueden ser cualesquiera de los 8 registros enteros (EAX...ESP)

581-La política de correspondencia de una memoria cache con la mitad de conjuntos que líneas es:

Asociativa por conjuntos de 2 vías



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

"vía" se refiere a cuántas "alternativas" (líneas) hay en un conjunto.
Además, todas las demás opciones tienen defectos

582-¿Cuál es el tamaño de la marca de caché en un ordenador capaz de direccionar 1 MB de memoria principal y 32 KB de memoria caché, que emplea un tamaño de palabra de 32 bits y correspondencia asociativa por conjuntos con 16 palabras por bloque y 8 bloques por conjunto, suponiendo que las direcciones de memoria utilizan 20 bits?

8 BITS

583-La ventaja de las transferencias de bloques en un bus es que sólo hay que transmitir la primera dirección y se ahorran las siguientes direcciones consecutivas.

Verdadero

584-Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.

Falso

585-En las arquitecturas del tipo registro-registro se usan instrucciones LOAD y STORE para transferir información entre registros y memoria principal.

Verdadero

586-El bus de direcciones contiene líneas para indicar el sentido de la transferencia de datos, por ejemplo una línea para distinguir entre lectura y escritura.

Falso

587-Si se necesitan 60 ns para escribir una palabra de datos de cache en memoria principal y cada bloque de cache tiene 8 palabras, ¿cuántas veces "seguidas" (sin que haya reemplazo) se tiene que escribir en un mismo bloque para que una cache de postescritura sea más eficiente que una de escritura inmediata?

No se puede responder con los datos proporcionados

588-Si el proceso de empaquetado de un producto (50 segundos de duración) puede segmentarse en 5 etapas, cada una de 10 segundos, de modo que 5 operarios puedan trabajar cada uno en una etapa, ¿cuál de las siguientes afirmaciones es falsa al aplicar la segmentación?

Cada 50 s saldrá un nuevo producto empaquetado, el mismo tiempo que cuando no había cadena de empaquetamiento

589-Son funciones de la unidad de control la lectura de memoria principal de la instrucción apuntada por el uPC y la codificación de esa instrucción.

Falso

590-Lo que se lee de una dirección de E/S siempre es lo mismo que lo último que se ha escrito en ella.

Falso

591-¿Cuál de las siguientes listas está correctamente ordenada temporalmente?

8088, 80386, Pentium, Pentium Pro,

a) Pentium 4, Pentium II.

X

b) 8088, 80386, Pentium, Pentium II, 80486, Pentium 4.

•

c) 8088, 80486, Pentium, Pentium Pro, Pentium II Pentium 4.

d) 80286, 8088, Pentium, Pentium Pro, Pentium III, Pentium 4.

592-En IA32, ¿cuál de las siguientes afirmaciones es incorrecta?

CALL ETIQUETA guarda en la pila la dirección de retorno y el registro de indicadores

593-Un ciclo de refresco en una memoria DRAM requiere un impulso RAS y un CAS.

FALSO

594-Un computador con 8 bits en el bus de direcciones puede direccionar como máximo:

256 palabras

595-¿Cuál de las siguientes afirmaciones es cierta?

Los modos de

- a) direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.



- b) El direccionamiento a registro es similar al directo. La diferencia es que el campo de direcciones referencia un registro en lugar de una dirección de memoria principal.
- c) En el direccionamiento directo, la dirección efectiva se encuentra en un registro de uso general del procesador.
- d) En el modo de direccionamiento implícito, el objeto direccionado es una constante contenida en la propia instrucción.

596-Un microprocesador es superescalar si tiene un número mayor de etapas y éstas son más cortas que las de un cauce ("pipeline") normal, permitiendo una velocidad de reloj mayor.

FALSO

597-Un mapa de direcciones de memoria es una representación pictórica del espacio de direcciones asignado a cada chip o tipo de memoria en el sistema.

VERDADERO

598-Se dice que las máquinas con arquitectura Von Neumann siguen un modelo de programa...

Almacenado.

599-¿Cuál de las siguientes afirmaciones respecto a la memoria RDRAM no es cierta?

El bus de datos suele ser muy ancho.

600-¿Cuál de las siguientes es una idea fundamental de la jerarquía de memoria?

Que dispositivos más pequeños y rápidos sirvan de cache para dispositivos más grandes y lentos

601-Sea un computador de 32 bits que dispone de una memoria cache de 512 KB y líneas de 64 bytes. ¿Cuántas líneas tiene la cache?

8192

602-Una placa madre de un 486 con un único SIMM de 30 contactos con 8 chips de 1M x 1, tiene 2 M palabras de memoria principal.

Falso

603-Con el repertorio IA32, para sumar %eax y %ebx dejando el resultado en %ecx se podría hacer lo siguiente:

lea (%eax, %ebx, 1), %ecx

604-El registro MDR/MBR...

contiene el valor que va a ser almacenado en la memoria, o bien se usa para recibir un valor procedente de la memoria

606-Respecto a requisitos de alineamiento de structs en gcc/IA32 x86 y x86-64, alguna de las siguientes afirmaciones es falsa

en x86-64 Linux alinea float a 8x (Windows también)

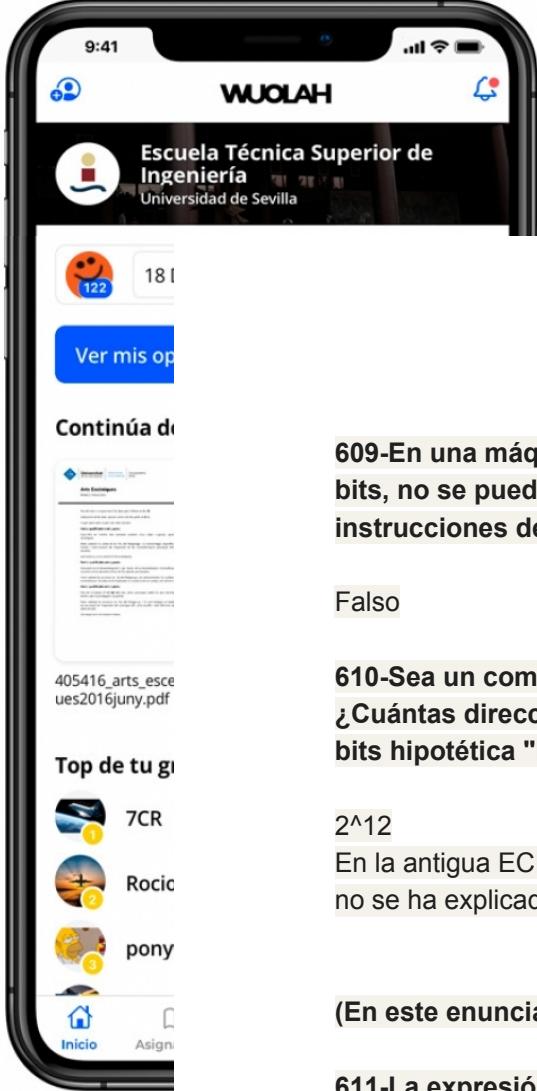
redacción original era "x86-64 Linux long double 8x, Windows no", motivo por el cual notamos errata transparencias (no dicen nada de x86-64 Windows, y dicen que x86-64 Linux long double 8x, cuando libro pág.325 dice que 16x). Posible corrección: "alinea long double a 12x (Windows no)". Redacción final es aún más claramente errónea.

607-¿En qué generación, dentro de la historia de los computadores digitales, aparece la microprogramación?

tercera

608-Respecto a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:

La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

609-En una máquina con 32 registros direccionables e instrucciones de 16 bits, no se pueden codificar 32 instrucciones de dos registros, 32 instrucciones de 1 registro y 32 instrucciones de 0 direcciones.

Falso

610-Sea un computador con 48 registros y 200 instrucciones máquina. ¿Cuántas direcciones de memoria permite el formato de la instrucción de 32 bits hipotética "beqz r1, r2, dir"?

2^{12}

En la antigua ECI se hacían estos cálculos, códigos Hamming, etc. En la nueva EC no se ha explicado con tanto detalle. Evitar esta pregunta.

(En este enunciado el símbolo \wedge representa potenciación)

611-La expresión que cumplen las direcciones de una memoria de bytes en las que queda mal alineada una palabra de 32 bits es Dirección mod 4 != 0.
VERDADERO

612-La instrucción NOP no se usa nunca, ya que no hace nada.

FALSO

613-Para que el modo de direccionamiento indirecto a través de registro coincida con el modo de direccionamiento indexado, el campo de desplazamiento en el direccionamiento indexado ha de ser igual al contenido del registro, pero cambiado de signo.

Falso

614-Respecto al sistema de Entrada / Salida, ¿cuál de las siguientes afirmaciones es incorrecta?

- a) La CPU se comunica con el periférico por medio del controlador y de software de E/S.
- b) Un protocolo sirve para “ponerse de acuerdo” en cosas como velocidad, paridad, nº de bits, etc.

- - c) Un controlador se encarga de la comunicación con la CPU.
 - d) La mayoría de los periféricos trabajan a velocidad muy superior a la CPU; por eso es necesario sincronizar.

615-En las transacciones de bloques del bus PCI se envía una dirección por cada dato, ya que direcciones y datos no están multiplexados en el tiempo.

Falso

616-La codificación Huffman utiliza menos bits para las instrucciones menos frecuentes y más bits para las instrucciones más utilizadas.

FALSO

617-La gestión distribuida del arbitraje consiste en un árbitro central del bus al que llegan múltiples líneas de petición de varios maestros potenciales y del que salen las correspondientes señales de concesión.

FALSO

618-Una memoria SRAM tiene una capacidad de 64 Kbits y utiliza 12 líneas para direccionamiento. Indique cuál es el tamaño de palabra de dicha memoria:

16 bits

619-Una dirección de memoria se refiere siempre a:

a)
una palabra

b)
16 bits

c)
un byte
 d)
ninguna de las anteriores

620-La idea fundamental en la que se basan los computadores RISC es minimizar el número de instrucciones necesarias para realizar una tarea determinada.

F

621-En la ejecución de una instrucción...

la ALU realiza las operaciones aritméticas y lógicas

622-¿Para qué se utiliza la función gettimeofday en la práctica de la "bomba digital"?

Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la contraseña o el PIN

623-Respecto a si un computador dispone de E/S independiente (separada) o usa E/S mapeada a memoria:

Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada

624- Compilar .s → ejecutable, usando sólo as y ld, sin gcc...

Se puede, usando en as y ld los modificadores (switches) que corresponda

625-Si AX = 0xFA50 y ejecutamos XOR \$0xFF, AX

Se realiza el complemento a 1 de AL.

626-Dado un camino de datos concreto, un posible formato de microprogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)

microbifurcaciones

627-En una arquitectura Von Neumann cada palabra de memoria contiene un campo que permite diferenciar entre instrucción o dato.

F

628-Se puede construir procesadores de varios anchos de palabra usando varios chips de sección de 4 bits.

V

629-El 8086 sólo permite direccionar directamente 640 KB de memoria. Por eso los PCs basados en este chip no disponían de más memoria RAM.

F

630-Al llamar a una función de 2 argumentos foo(arg1, arg2), ¿cuál es el orden correcto de las operaciones? (suponiendo convención de llamada x86 cdecl, y que foo requiere ajustar marco de pila, esto es, salvar %ebp)
push arg2, push arg1, call foo, push %ebp

631-En la E/S controlada por interrupciones:

La CPU transfiere el control a una rutina de servicio cuando recibe una interrupción.

632-Los computadores con estructura Von Neumann utilizan el modelo de programa externo, frente a programa cableado y almacenado.

F

633-Un computador que utilice el sistema big-endian, almacena el número 2143h a partir de la dirección 0 como:

21h en el byte de la dirección 0 y 43h en el byte de la dirección

1

634-La microprogramación vertical se caracteriza por tener:
muchas codificaciones

635-Se dispone de un computador cuyo tiempo medio de acceso al sistema de memoria caché y memoria principal es de 18 ns. Si la tasa de fallo de la caché es de 0,2 y el tiempo de acceso a la memoria principal es 50 ns. ¿Cuál es el tiempo de acceso a la caché?

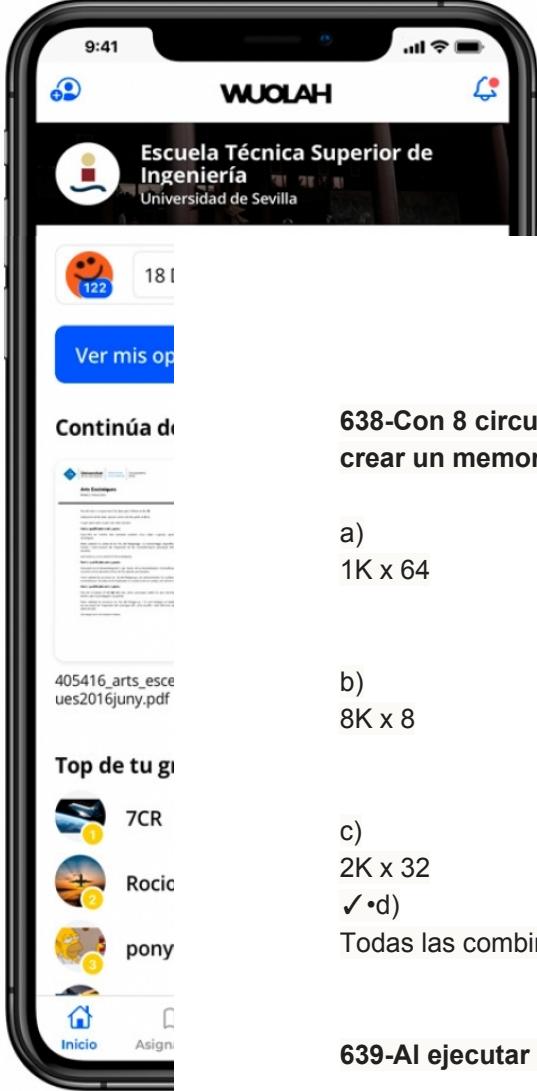
8 ns

636-Siempre que se utiliza "polling" la prioridad de los dispositivos interruptores queda fijada mediante encadenamiento ("daisy-chain").

Falso

637-¿Cuál de las siguientes afirmaciones es cierta?

El modo de direccionamiento indirecto a través de registro es el modo más sencillo de direccionamiento a memoria.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

638-Con 8 circuitos de memoria RAM de 1K x 8 se puede crear un memoria de:

- a)
1K x 64

- b)
8K x 8

- c)
2K x 32
- ✓ d)
Todas las combinaciones anteriores son posibles

639-Al ejecutar el fragmento de código:

```
leal -48(%eax), %edx
cmpl $9, %edx
ja .L2
```

se salta a .L2 si el contenido del registro %eax:

está fuera del intervalo [48,57]

640-Cada bit de una RAM dinámica ha de ser refrescado cada pocos nanosegundos.

Falso

641-Cuál de los siguientes no es un modo de direccionamiento IA32?

- a)
Cache
no existe, ver [T2.1.3ConASM] tr.27
- b)

Registro

c)
Inmediato

d)
Memoria

642-¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC es cierta?

Ninguna de las otras afirmaciones es cierta

643-Si se dice que en un sistema computador cada dirección especifica uno o dos puertos de E/S, se refiere a que:

Un puerto será de sólo lectura, otro de sólo escritura, y ambos se decodifican en la misma dirección

644-Un bus se compone de:

Líneas de control/estado, líneas de dirección y líneas de datos

645-El Intel 8086:

Incluía instrucciones de multiplicación.

646-El fragmento de código:

```
poll: in a, 0x20
      cmp a, $0
      jnz poll
      load a, 0x11
      out 0x21, a
```

corresponde a:

E/S programada con consulta de estado

647-La idea de la arquitectura RISC se debe, entre otros, a:

John Cocke

648-¿Qué necesitamos para construir una memoria de 1K x 8 bits?

8 memorias de 256 x 4 bits y un decodificador de 2 a 4

649-¿Cuál de los siguientes grupos de instrucciones sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

CMP, TEST

650-Un TLB (buffer de traducción anticipada) tiene un número de entradas o elementos mucho menor que el número de páginas de la memoria virtual.

V

651-En una arquitectura little-endian el código de operación aparece al final de la instrucción.

F

652-Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señalar la respuesta falsa)

cola de instrucciones (precaptación)

653-Nunca puede utilizarse "polling" para identificar la fuente de una interrupción en un sistema con "daisy-chain"

F

654-SCSI son las siglas de Small Computer System Interface (interfaz del sistema para computadores pequeños).

Verdadero

655-Un procesador cuya instrucción CALL guardara la

dirección de retorno en un registro RL (llamado "de enlace"):

No permitiría llamadas anidadas ni recursivas.

656-En un procesador RISC, las bifurcaciones no degradan las prestaciones del "pipeline".

F

657-Una de las funciones de una interfaz de E/S es generar señales de temporización cuando se necesiten, según el esquema que se utilice el subbus de control.

658-¿Cuál afirmación es FALSA al comparar las arquitecturas x86 y x86-64?

659-A medida que aumenta el tamaño de página en un sistema de memoria virtual, ¿qué ocurre con el tamaño de las tablas de páginas?

Disminuye

660-¿En qué tipo de memoria virtual es un problema la fragmentación externa?

661-Se tiene una memoria que emplea entrelazado. Si fallan varias celdas contiguas de uno de sus chips de memoria, ¿con qué tipo de entrelazado de memoria sería más fácil poder utilizarla?

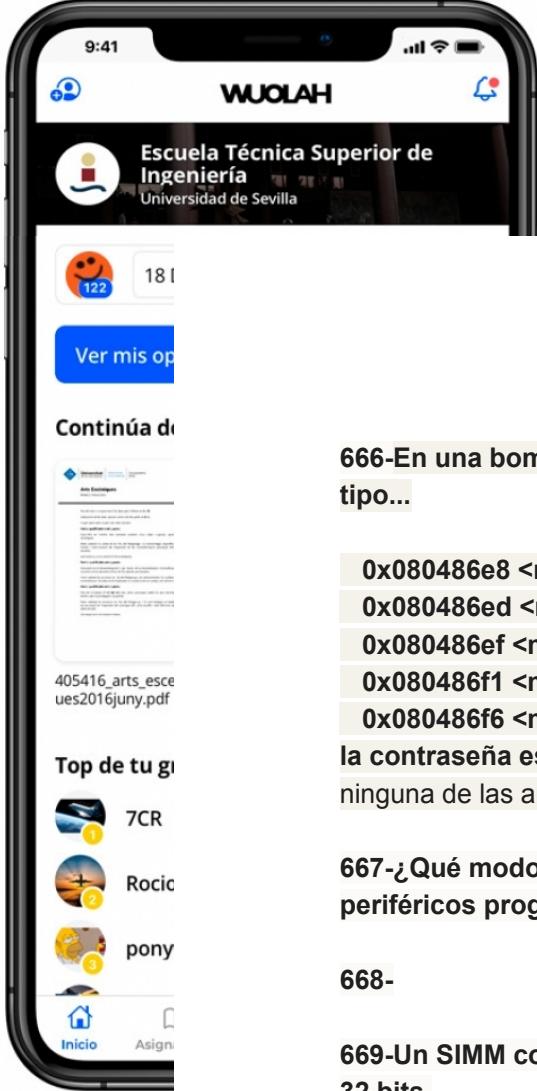
662-Una cache de 256 B asociativa por conjuntos de 4-vías con líneas de 16 B tendría

4 conjuntos

663-Para diseñar una memoria con ancho de palabra $k \cdot m$ (y mismo nº palabras que los módulos) a partir de módulos con ancho de palabra m , se utilizan k módulos (mediante _ se representa subíndice)

664-El Pentium III es superescalar porque permite "emitir" varias instrucciones por ciclo de reloj.

665-¿Cómo actúa el indicador N del registro de indicadores de estado?



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

666-En una bomba como las estudiadas en prácticas, del tipo...

```
0x080486e8 <main+120>: call 0x8048524 <strncmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
0x080486f6 <main+134>: ...
```

la contraseña es...

ninguna de las anteriores

667-¿Qué modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?

668-

669-Un SIMM con 8 chips de 4Mx1 contiene 1 M palabras de 32 bits.

F

670-En el contexto de las DRAM, la activación de la señal CAS se emplea para realizar ciclos de refresco.

671-La "postescritura ("write-back") marcada" es más eficiente que la "postescritura siempre".

V

672-La cache es gestionada por:
unidades de "manejo" (gestión) hardware

673-Un modo de direccionamiento en el que se especifica un registro y una dirección de memoria cuyo contenido se suma al contenido del registro base para obtener la dirección efectiva, se conoce como:

- a)base con desplazamiento
- b)absoluto o directo
- c)indirecto a registro a través de memoria
- d)ninguna de las anteriores*****

674-Si la estructura struct a ocupa un espacio de 28 bytes

en memoria, ¿cuántos bytes ocupa la siguiente estructura struct b cuando se compila en 64 bits?

```
struct b {  
    struct a a1;  
    int i;  
    struct a a2;  
}  
60 bytes
```

675-En un sistema de memoria con entrelazado de orden inferior y M módulos de memoria la dirección de memoria a está en el módulo a mod M.

V

676-Alguna de las siguientes señales no sirve de entrada a la unidad de control. ¿Cuál?

contador de programa (bits del registro PC)

677-¿Cuál es la característica tecnológica principal de la tercera generación de computadores?

Los circuitos integrados

678-¿Cuál de las siguientes secuencias de instrucciones multiplica el (contenido del) registro EAX por 18?

```
leal (%eax,%eax,8), %eax  
leal (%eax,%eax), %eax
```

679-Un sistema no segmentado tarda 20 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 6 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se aproxima a:

3,33

680-Al traducir la sentencia C r->i = val; gcc genera el código ASM movl %edx, 12(%eax). Se deduce que

el desplazamiento de i en *r es 12
y en concreto r está en %eax

681-El computador EDVAC, propuesto por John Von Neumann, presentaba dos importantes diferencias respecto al ENIAC:

Empleaba aritmética binaria y permitía trabajar con un programa almacenado.

682-Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?

31,25 microsegundos

683-El lenguaje máquina...

a)

es un conjunto de nombres simbólicos o nemotécnicos.

b)

facilita la portabilidad de los programas.

c)

es el mismo para todos los computadores.

✓ •d)

Ninguna de las respuestas anteriores es correcta.

684-La instrucción negl:

realiza el complemento a dos

685-Una caché totalmente asociativa es equivalente a una asociativa por conjuntos de una vía.

Falso

686-En la secuencia de programa siguiente:

804854e:e8 3d 06 00 00 call 8048b90 <main>

8048553:50 pushl %eax

¿cuál es el valor que introduce en la pila la instrucción call?

0x8048553

687-Una máquina que almacene el número 4321h a partir de la dirección 0 como 21h, 43h utiliza el sistema big-endian.

Falso

688-Las instrucciones que asignan la dirección 0x78e00 a un puntero situado en la posición 0xff00 son:

mov \$0x78e00,%eax

mov %eax,0xff00

689-En general, un operación segmentada ("pipelined") requiere el mismo tiempo o más, desde el principio hasta el fin y para un único par de operandos, que la misma operación en una implementación no segmentada.

V

690-La penalización por una falta de página suele ser de cientos de miles de ciclos de reloj de la CPU.

Verdadero

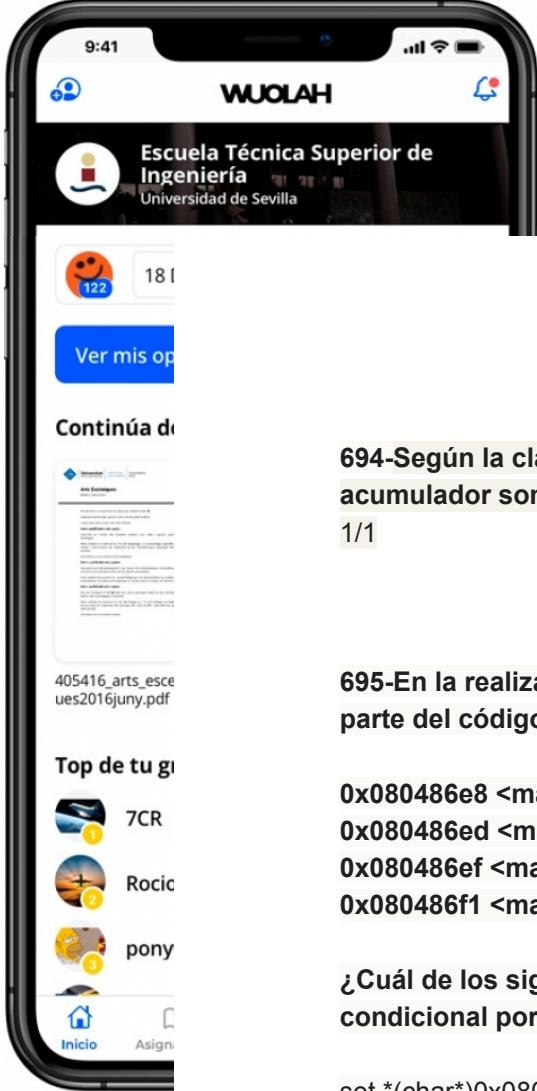
691-En el direccionamiento inmediato el operando reside en:

en la instrucción tras el código de operación

692-El bus del sistema es

el que conecta CPU-M, ya sea un sistema con bus único o con múltiples buses

693-La CPU guarda automáticamente el contexto del programa que ejecuta cuando le llega y atiende una petición de DMA.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

694-Según la clasificación m/n, las máquinas de acumulador son de tipo

1/1

695-En la realización de la práctica de la bomba digital, una parte del código máquina es el siguiente:

```
0x080486e8 <main+120>: call 0x8048524 <strcmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
```

¿Cuál de los siguientes comandos cambiaría el salto condicional por un salto incondicional?

set *(char*)0x080486ef=0xeb

696-Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más fácil descubrir la(s) contraseña(s)? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" se entiende aplicar la cifra del César (sumar o restar una constante fija a los códigos ASCII).

1 string del fuente se cifra, y se compara con el string del usuario

aunque hubieran sido 2, aunque su hubieran cifrado y concatenado o al revés, sigue siendo igual de fácil, porque con el debugger se puede ver qué hay que poner de contraseña literalmente

697-Un archivo .o que contiene código objeto:

Contiene instrucciones máquina binarias.

698-¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?

En las transferencias asíncronas

699-La instrucción MOVSB del 8086 transfiere el byte apuntado por ES:DI al acumulador AL.

FALSO

700-En el bus PCI las direcciones y los datos están multiplexados en el tiempo, y existen transacciones de bloques de datos en las que se envía una única dirección y se envían (o reciben) muchos datos.

Verdadero

701-¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, volverá a ser referenciado pronto"?

Localidad temporal

702-Después de ejecutar una instrucción de suma sobre dos números con signo de la que sabemos que no provocará overflow (los dos números son pequeños en valor absoluto), queremos comprobar si el resultado de la suma es menor que 0. ¿Qué flag necesita comprobar la instrucción de salto condicional equivalente a “if (resultado<0) then goto label”?

SF

703-Un bus de ciclo partido está disponible para otro maestro en las ranuras entre la petición de una lectura y la contestación con el dato.

VERDADERO

704-En una caché con 64 bytes de longitud de línea, ¿qué bits de una dirección de memoria se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección?

Los 6 bits menos significativos

705-La memoria caché del computador es:

a)

Más rápida que la memoria principal

b)

De menor capacidad que la memoria principal

✓ •c)

a) y b) son correctas

d)

Ninguna de las anteriores es correcta

706-Una jerarquía de memoria consta de una cache de con una tasa de aciertos del 92% y 4 ns de tiempo de acceso y una memoria principal con una tasa de aciertos del 100% y 100 ns de tiempo de acceso. ¿Cuál es el tiempo promedio estimado de acceso a memoria?

12 ns

$0.92 \cdot 4\text{ns} + 0.08 \cdot (100+4)\text{ns}$

707-El bus GPIB utiliza handshake de 6 flancos. Esto es necesario para que un emisor pueda enviar datos a varios receptores distinguiéndose el hecho de que el dato haya sido leído por todos del hecho de que todos estén listos para recibir un nuevo dato.

VERDADERO

708-Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

EAX

709-Para la memoria virtual se suele utilizar correspondencia directa

FALSO

710-Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:

compilador

711-¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una función long int al terminar ésta?

- a) La instrucción RET lo almacena en un registro especial de retorno.
- b) Por convención se guarda en %eax.
- c) Se almacena en pila justo encima de los argumentos de la función
- d) **CORRECTA->**Ninguna de esas formas es la correcta

712-¿Qué política de memoria virtual para colocar nuevos segmentos en los huecos libres de la memoria principal evita el que se generen huecos pequeños?

Peor ajuste

713-¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

2

714-El tiempo de acceso de un SIMM es mayor que el de un único chip, debido al retraso de los decodificadores que contiene para generar los distintos "Chip Select".

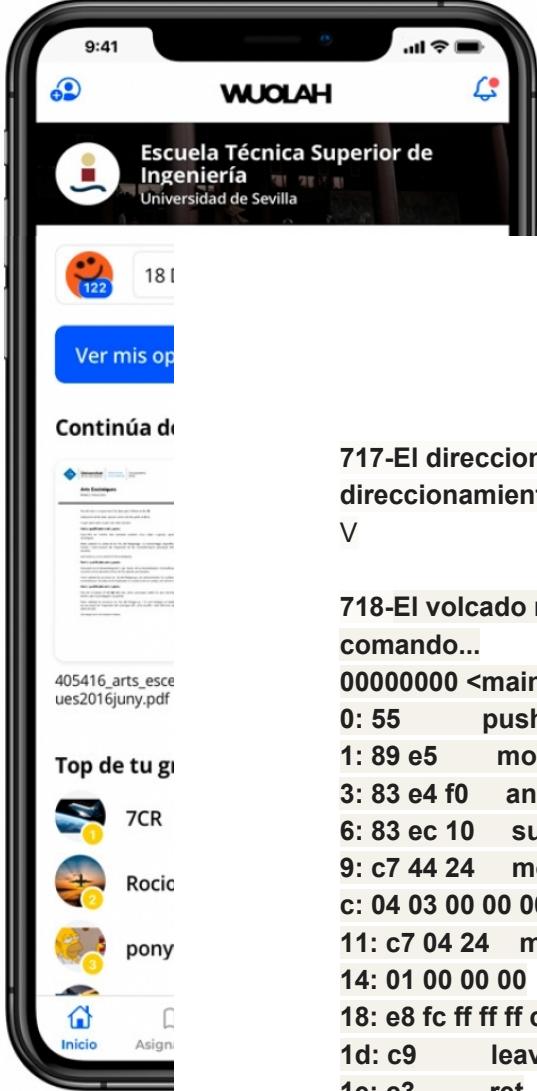
FALSO

715-Usar un procesador más rápido siempre implica un incremento proporcional en las prestaciones de un computador, incluso si la velocidad de la memoria principal permanece inalterada.

FALSO

716-El resultado de desplazar aritméticamente dos posiciones hacia la derecha el número de 8 bits en complemento a dos –32 es:

-8



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Continúa di

717-El direccionamiento inmediato no es más lento que el direccionamiento directo.

V

718-El volcado mostrado abajo se ha obtenido con el comando...

00000000 <main>:

```
0: 55      push %ebp
1: 89 e5    mov %esp,%ebp
3: 83 e4 f0  and $-16,%esp
6: 83 ec 10  sub $0x10,%esp
9: c7 44 24  movl $3, 4(%esp)
c: 04 03 00 00 00
11: c7 04 24  movl $0x1,(%esp)
14: 01 00 00 00
18: e8 fc ff ff call <main+0x19>
1d: c9        leave
1e: c3        ret
```

objdump -d p2.o

-d porque es un desensamblado (no cabeceras -h ni tablas -t)

719-La conexión entre un dispositivo de E/S y el procesador mediante bus:

Permite conectar en paralelo varios dispositivos

720-Según el concepto de máquina virtual de Tanenbaum, el nivel de máquina convencional consiste en una computadora hipotética con un determinado lenguaje máquina, cuyos programas son interpretados por la máquina virtual del nivel de microprogramación, supuesto que éste existe.

Verdadero

721-Si desplazamos a la izquierda un registro 3 posiciones:

Lo multiplicamos por 8

722-Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,  
      0x10000000, 0x20000000  
longlista: .int (. -lista)/4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\n\0"
```

```
push resultado+4  
push resultado  
push resultado+4  
push resultado  
push $formato  
call printf  
add $20, %esp  
Sí.  $20 = 8(\%llx) + 8(\%llu) + 4(\$formato)$ 
```

723-Alguno de los siguientes no es un nombre de registro en una máquina IA32 en modo 32 bits

sil
Sí lo sería en modo 64 bits

724-Las instrucciones máquina contienen toda la información necesaria para ejecutarse, y además su interpretación es independiente de la posición que ocupan en el programa.

Verdadero

725-Para realizar la microoperación MAR <- PC, habrá que activar:

EnPC y LdMAR

726-El primer nivel de una jerarquía de memoria tiene una tasa de aciertos del 75% y las peticiones de memoria tardan 12 ns en completarse si dicha posición se encuentra en ese nivel y 100 ns si no es así. ¿Cuál es el tiempo medio de acceso de la jerarquía?

34 ns

727-¿En qué registro se pasa el primer argumento a una función en Linux gcc x86-64?

edi

728-¿Cuál de las siguientes secuencias de instrucciones multiplica %eax por 10?

a)

leal(%eax,%eax,4), %eax
sal \$2, %eax
sería $x5x4=x20$

b)

imull \$0x10, %eax
sería x16

c)

addl %eax, %eax
shll \$5, %eax
sería $x2x32=x64$

✓ •d)

Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una

729-¿Qué política de colocación en caché necesita más comparadores, la correspondencia asociativa por conjuntos o la correspondencia por sectores?

Depende de si es mayor el número de bloques por conjunto o el número de sectores

730-Dada la siguiente definición de datos:

lista: .int 0x10000000, 0x50000000,
 0x10000000, 0x20000000

longlista: .int (.lista)/4

resultado: .quad 0x123456789ABCDEF

formato: .ascii "suma=%llu=%llx hex\n\0"

la instrucción movl longlista, %ecx copia el siguiente valor:

4

Práct.2, Tut, pág.7

731-¿Cuál de las siguientes características es típica de la microprogramación horizontal?

Ninguna o escasa codificación.

732-HIJO DE PUTA

733-En una memoria con entrelazado de orden inferior, M módulos y acceso simultáneo, el tiempo de acceso a las $2M$ palabras cuyas direcciones son $0, 1, \dots, 2M-1$ es menor o igual que $3Ta$, siendo Ta el tiempo de acceso a cada módulo de memoria.

Verdadero

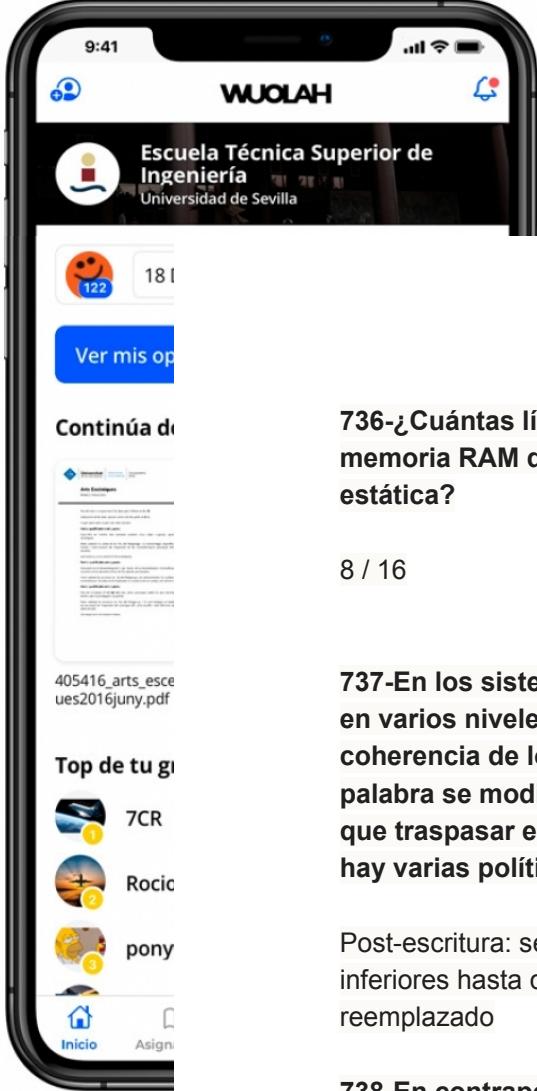
734-Los buses síncronos y semisíncronos disponen de una línea de reloj "maestro", y todas las actividades del bus tienen una duración igual a un número entero de ciclos de ese reloj.

Verdadero

735-Suponiendo que todos los registros inicialmente contienen el valor 1, ¿cuál es el valor de ECX tras la ejecución de la siguiente secuencia de instrucciones?

```
mov $4, %eax  
mov $3, %ebx  
lea (%eax, %eax), %ecx  
sub %ebx, %ecx  
imul %ecx, %eax
```

5



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

736-¿Cuántas líneas de dirección son necesarias en un memoria RAM dinámica de 64 K palabras? ¿Y en una estática?

8 / 16

737-En los sistemas con una jerarquía de memoria dividida en varios niveles se da el problema de la consistencia o coherencia de los datos entre los distintos niveles. Si una palabra se modifica en un nivel, en algún momento habrá que traspasar ese cambio a los niveles inferiores. Para ello hay varias políticas:

Post-escritura: se retrasa la actualización en los niveles inferiores hasta que el bloque modificado tenga que ser reemplazado

738-En contraposición a un ejecutable Linux ELF, un fichero objeto (obtenido con gcc -c)

ubica el código (y los datos) a partir de la posición 0x0, las direcciones definitivas sólo se calculan tras enlazar

739-Un Pentium 4 a 3,2 GHz dispone de 7 unidades de ejecución en paralelo, con 20 etapas de segmentación, y es capaz de emitir (comenzar a ejecutar) 3 instrucciones en cada ciclo de reloj. ¿Qué velocidad aproximada de ejecución de instrucciones será capaz de alcanzar (MIPS = millones de instrucciones por segundo)?

9000 MIPS

740-En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?

mov \$-1, %edi
mov no afecta a los flags

741-Un sistema no segmentado tarda 200 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce segmentado de 20 etapas con un ciclo de reloj de 12 ns. Cuando se procesan muchas tareas, la máxima ganancia de velocidad que podría obtenerse se acerca a:

16,67

742-Las memorias caché con política de extracción anticipativa (prebúsqueda o preextracción) no soportan búsqueda por demanda.

Falso

743-Si el tiempo de acceso a la memoria cache es de 2 ns y el tiempo necesario para tratar un fallo de cache es de 80 ns, ¿cuál es la tasa de aciertos necesaria para que el tiempo medio de acceso al sistema de memoria sea de 10 ns?

0.9

744-La cache con correspondencia directa se puede considerar como un caso límite de la asociativa por conjuntos, en donde...

745-Un overflow nunca puede ocurrir cuando:

•d)
se suma un número positivo a un número negativo
resultado queda entre ambos => se puede representar

746-El algoritmo LRU ("Menos Recientemente Usado/a") se puede utilizar tanto en memoria caché como en sistemas de memoria virtual con paginación.

Verdadero

747-En una CPU de 32 bits con memoria de bytes, el problema es que.

Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU

748-La primera instrucción ensamblador de una subrutina compilada con gcc en Linux/x86 cdecl suele ser:

push %ebp

749-¿En qué generación, dentro de la historia de los computadores digitales, aparecieron la microprogramación, la segmentación de cauce, la memoria cache, los S.O. multiusuario y la memoria virtual?

3^a generación (1965-75)

en la 3^a generación se inventó casi todo

750-Un cauce ("pipeline") de instrucciones con 5 etapas tarda 7 ciclos de reloj o más en ejecutar 3 instrucciones si éstas utilizan las cinco etapas.

Verdadero

751-PC y SP son dos registros de uso general (GPR).

Falso

752-Si se necesitan 60 ns para escribir una palabra de datos de caché en memoria principal y cada bloque de caché tiene 8 palabras, ¿cuántas veces seguidas se tiene que escribir en un mismo bloque para que una caché de postescritura sea más eficiente que una de escritura inmediata?

Depende de la tasa de aciertos

753-¿Cuál de las siguientes afirmaciones acerca de las

interrupciones en el PC (modo real) es cierta?

a)

Cada vector de interrupción es una palabra de 16 bits

b)

Existen 1024 vectores de interrupción

c)

La tabla de interrupciones tiene un tamaño de 256 bytes

• d)

Todas las interrupciones se pueden generar por software

PREGUNTAS TEST EC TEMA 3

1. ¿Cómo actúa el indicador de signo?

- a) Se pone a 1 cuando el resultado es positivo
- > b) Se pone a 1 cuando el resultado es negativo
- c) Se pone a 1 cuando el resultado es distinto
- d) Se pone a 0 cuando el resultado es negativo

2. La microprogramación vertical se caracteriza por tener:

- a) escaso o ningún solapamiento entre campos
- b) capacidad para expresar un alto grado de paralelismo en las microoperaciones a ejecutar
- c) microinstrucciones largas
- > d) mucha codificación

3. Un procesador con una unidad de control microprogramada tiene una memoria de control de 256 palabras de 16 bits, de las que 128 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?

- > a) 256 bits
- b) No se produce ahorro
- c) 4096 bits
- d) 3840 bits

4. La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...

- > a) dos multiplexores de 2 a 1
- b) tres conexiones directas al bus común
- c) tres demultiplexores
- d) dos buffers triestado

5. Sea un formato de microinstrucción que incluye dos campos independientes de 9 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

- a) 1
- b) 4
- > c) 8
- d) 9

6. ¿Cómo actúa el indicador N del registro de indicadores de estado?

- > a) Se pone a 1 cuando el resultado es negativo.
- b) Se pone a 1 cuando el resultado es positivo.
- c) Se pone a 1 cuando el resultado de una operación es 0.
- d) Se pone a 0 cuando el resultado es negativo.

7. Los procesadores comerciales con unidad de control microprogramada suelen almacenar los micropogramas...

- a) en una RAM.
- b) en un banco de registros.
- > c) en una ROM.
- d) en una PLA.

8. Respecto a MBR y MAR

- a) Ambos permiten guardar información sobre el marco de pila //sin relación directa
- b) Ambos son accesibles por el programador //ninguno
- c) MAR contiene el dato/instrucción que se leerá o escribirá en memoria //ese es MBR
- > d) MAR requiere menos señales de control que MBR //en Tema 3 tr.10, sólo una (LoadMAR),

//frente a 4 de MBR (Load/Enable Mem/Bus)

9. Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?

- a) $k + n \cdot 2^m$
- b) $k + n^m$
- > c) $k + n \cdot (2^m - 1)$
- d) Ninguno de los anteriores

10. Un computador tiene una memoria de control de 640 palabras de 70 bits, de las que 280 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación en lugar de microprogramación?

- >a) 19440
- b) 42280
- c) 9840
- d) ninguno de los anteriores resultados es exacto.

11. Un computador tiene una memoria de control de 16000 palabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorramos usando nanoprogramación en lugar de microprogramación?

- > a) 3744250
- b) 259206
- c) 287935
- d) ninguno de los resultados anteriores es exacto

12. En un camino de datos con un solo bus, para realizar la operación de copia de un registro r1 en un registro r2, es decir $r2 \leftarrow r1$, es necesario:

- a) Activar la carga del registro r1 y habilitar la salida triestado del registro r2
- b) Habilitar la salida triestado del registro r2 y activar la carga de los registros r1 y r2
- c) Habilitar las salidas triestado de los registros r1 y r2 y activar la carga del registro r2
- > d) Habilitar la salida triestado del registro r1 y activar la carga del registro r2

13. Respecto a las unidades de control nanoprogramadas:

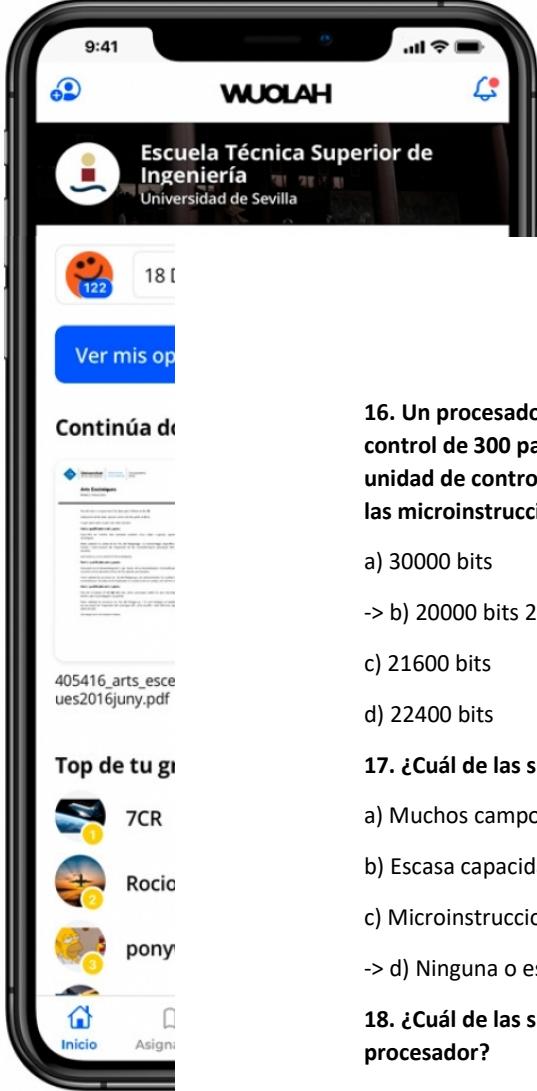
- > a) La anchura de la memoria de nanoprograma es la misma que la de memoria de microprograma en un diseño de la misma unidad de control que no usara nanoprogramación.
- b) La realización nanoprogramada de una unidad de control es más rápida que la microprogramada.
- c) El diseño de las unidades de control nanoprogramadas debe ser vertical.
- d) Suponiendo una memoria de microprograma con n microinstrucciones de w bits cada una, de las cuales 2^m son distintas, el ahorro en bits si se utiliza nanoprogramación es $(n \cdot m + 2^m \cdot w) - n \cdot w$.

14. En el pseudocódigo usado para representar las microinstrucciones, la expresión “`goto f(IR)`”:

- a) Permite saltar a la dirección de memoria de control del principio de un microbucle.
- b) Se utiliza para realizar un microsalto condicional en función del registro de estado.
- c) Realiza una llamada a una microsubrutina.
- > d) Salta a una dirección de memoria de control que depende de la instrucción máquina actual.

15. Para conectar las salidas de dos registros hacia un bus común en el datapath...

- > a) se pueden usar dos buffers triestado.
- b) no se puede usar un multiplexor.
- c) se puede realizar una conexión directa.
- d) se puede usar un demultiplexor.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

16. Un procesador con una unidad de control microprogramada tiene una memoria de control de 300 palabras de 100 bits, de las que 200 son diferentes. Si se rediseñara como unidad de control nanoprogramada, ¿qué tamaño ocuparía la nanomemoria que contiene las microinstrucciones completas sin repeticiones?

- a) 30000 bits
- > b) 20000 bits 200 uinstr. x 100 bits
- c) 21600 bits
- d) 22400 bits

17. ¿Cuál de las siguientes características es típica de la microprogramación horizontal?

- a) Muchos campos solapados.
- b) Escasa capacidad para expresar paralelismo entre microoperaciones.
- c) Microinstrucciones cortas.
- > d) Ninguna o escasa codificación.

18. ¿Cuál de las siguientes funciones no corresponde a la unidad de control de un procesador?

- > a) Calculo de operaciones de coma flotante
- b) Decodificación de las instrucciones
- c) Secuenciamiento de las instrucciones
- d) Generación de las señales de control que provocan la ejecución de cada instrucción

19. Para el procesador con unidad de control microprogramada estudiado en clase, Tanenbaum define 15 codops de 4bits para instrucciones con un operando dirección de 12bits (LODD M, STOD M, ADDD M, ...), y en lugar de gastar el último codop 1111 en otra instrucción con una dirección, realiza una extensión de codop de 3bits para definir ocho instrucciones más. En general esas instrucciones no tienen ningún operando, salvo INSP Y (sp+=Y) y DESP Y (sp-=Y), que tienen un operando inmediato.

- a) Y es un operando de 8 bits y no sobra espacio de codificación (huecos) para modificaciones sin aprovechar bit8

-> b) Y es un valor de 8 bits pero podría definirse que fuera de 9 bits sin ninguna dificultad

- c) Y es una dirección corta de 9 bits y podría definirse que fuera de 10 u 11 bits cambiando la codificación por extensión

// 11bits imposible, sólo cabrían INSP/DESP, no habría sitio para PUSH/POP etc

- d) Y ocupa 10 bits, pero podría ser de 11 bits cambiando la codificación por extensión y uniendo ambas instrucciones en una sola ADDSP Y (sp+=Y) en donde Y se interpretara como número con signo en complemento a dos

// Y no ocupa 10bits.

//Sí que sería posible ADDSP codop 1111 1yyy yyyy yyyy con 11bit para Y, y entonces codop //1111 0ccc 0000 0000 permite de sobra los 6 codops restantes. [T3.4Tnbaum] tr.79

20. Una unidad de control microprogramada con secuenciamiento explícito con dos direcciones por microinstrucción, tiene una memoria de control con 35 bits de longitud de palabra. Si las microinstrucciones emplean 15 bits en total para los campos de control y de tipo y condición de salto, el número máximo de palabras de la memoria de control de esta unidad de control microprogramada es de:

- a) 10
- b) 2^{20}
- > c) 2^{10}
- d) 20

21. ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?

- > a) Una memoria.
- b) Un registro.
- c) Un multiplexor.
- d) Un contador.

22. En una arquitectura RISC típica:

- a) no puede usarse segmentación.
- b) la programación resulta mucho más simple que en una arquitectura CISC.
- > c) se usa un porcentaje elevado de las instrucciones del repertorio.
- d) la UC es más compleja que en una arquitectura CISC.

//Evitar esta pregunta (de la antigua titulación), las transparencias que hablan de RISC están //ahora repartidas por los distintos temas

23. Sea un formato de microinstrucción que incluye dos campos independientes de 10 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

- a) 14
- b) 13
- c) 10
- >d) 9

24. La salida de un campo del registro de microinstrucción que solapa dirección de salto y algunas señales de control han de conectarse a:

- a) una ROM o PLA
- >b) un demultiplexor controlado por el tipo de salto
- c) el registro de instrucción
- d) la memoria de control

25. Una unidad de control microprogramada se denomina "con secuenciamiento de microinstrucciones explícito" según tenga o no tenga

- a) ROM/PLA para traducir el codop en dirección de inicio de microprograma (goto f(IR))
//todos los diseños implícitos en las transparencias usan ROM traducción, pero también se
//podría añadir ROM al explícito
- b) microcódigo de decodificación que analice el codop bit a bit de izquierda a derecha
//con ROM traductora (goto f(IR)) no haría falta ese análisis
- > c) micro-contador de programa atacando a las líneas de dirección de la memoria de control
// si y sólo si tiene micro-PC, sería implícito
- d) un multiplexor para seleccionar la fuente de la dirección de la memoria de control
//todos los diseños en las transparencias tienen un MUX en la dirección de la memoria de
//control (o en el micro-PC)

26. ¿Cuál de las siguientes afirmaciones es verdadera?

- > a) La unidad de control necesita como entrada el registro de estado para poder controlar la ejecución de las instrucciones de salto condicional.
- b) El registro puntero de pila es un registro de propósito general que suele contener tanto direcciones como datos.
- c) Las únicas instrucciones en las que algunas de sus fases de ejecución llevan un acceso a memoria son las instrucciones load y store.
- d) El registro de instrucción es un registro de propósito específico que contiene la dirección de la siguiente instrucción a ejecutar.

27. En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de

- a) 17 bits
- b) 16 bits
- > c) 5 bits
- d) 4 bits

28. Para realizar la microoperación MAR <- PC, habrá que activar:

- a) EnPC y EnMAR
- b) LdPC y EnMAR
- > c) EnPC y LdMAR
- d) LdPC y LdMAR

29. Una posible codificación en microinstrucciones de la instrucción CALL X es:

- a) SP=SP-1 ; m[SP]=PC ; PC=PC+1
- > b) SP=SP-1 ; m[SP]=PC ; PC=X
- c) PC=X ; SP=SP-1 ; m[SP]=PC
- d) SP=PC-1 ; m[SP]=PC ; PC=X

30. Dado un camino de datos concreto, un posible formato de microprogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)

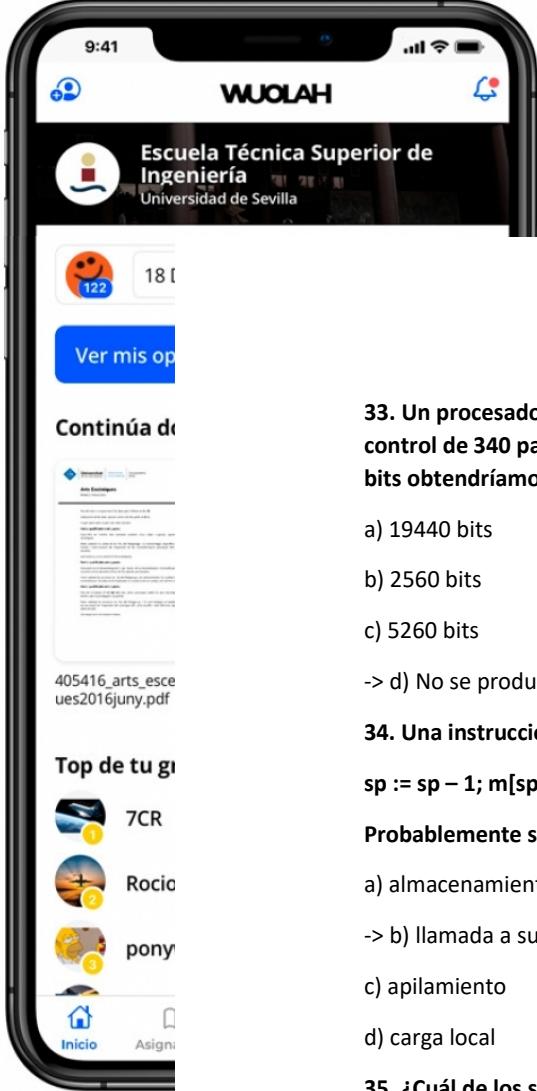
- a) longitud relativa de microinstrucción
- > b) microbifurcaciones
- c) solapamiento
- d) codificación

31. Alguna de las siguientes *no* es una operación básica de la unidad de control:

- a) realizar operación ALU y guardar resultado en registro
- b) transferir un registro a otro
- > c) (guardar o recuperar) un registro (en / de) la pila
- d) (leer o escribir) un registro (de / a) memoria

32. Un computador tiene una memoria de control de 16 Kpalabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorramos usando nanoprogramación en lugar de microprogramación?

- a) 3928652
- b) 259206
- c) 287935
- > d) ninguno de los resultados anteriores es exacto



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

33. Un procesador con una unidad de control microprogramada tiene una memoria de control de 340 palabras de 16 bits, de las que 180 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?

a) 19440 bits

b) 2560 bits

c) 5260 bits

-> d) No se produce ahorro

34. Una instrucción máquina puede desglosarse en las siguientes operaciones elementales:

$sp := sp - 1; m[sp] := pc; pc := x$

Probablemente se trate de una instrucción de:

a) almacenamiento local

-> b) llamada a subrutina

c) apilamiento

d) carga local

35. ¿Cuál de los siguientes grupos de señales son entradas a la unidad de control?

a) Las señales de carga/incremento/desplazamiento de registros

b) Las señales de selección de entradas de multiplexores del datapath

-> c) Los bits del registro de indicadores (flags)

d) Los bits de las opciones b y c

36. Respecto a los términos microinstrucción y microcódigo:

a) Son equivalentes, llamamos microcódigo o microinstrucción a una palabra de la memoria de control

b) Una microinstrucción está programada en microcódigo, que es un lenguaje para programar señales de control

c) Un microcódigo controla una serie de señales de control relacionadas (por ejemplo, el código 000 para que la ALU realice la suma), y varios juntos forman una microinstrucción

-> d) Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoria

37. El control residual se utiliza para:

a) reducir el tiempo de ejecución de las instrucciones máquina

b) eliminar los bits residuales de la ejecución de los microinstrucciones

-> c) reducir el tamaño de la memoria de control

d) ninguna de las anteriores es cierta

38. Son funciones de la unidad de control:

- a) la codificación de las instrucciones máquina
- b) la lectura de memoria principal de la instrucción apuntada por el µPC
- > c) el secuenciamiento de las instrucciones máquina
- d) todas las respuestas son ciertas

39. En cuanto al control microprogramado:

- a) se guardan en una ROM las instrucciones máquina del conjunto de instrucciones.
- b) la UC se construye con puertas lógicas, biestables, etc.
- > c) se usa en CISC para facilitar el diseño de la UC tan compleja.
- d) la lentitud en la ejecución de las instrucciones máquina la impone directamente la tecnología hardware usada.

40. Alguna de las siguientes señales no sirve de entrada a la unidad de control. ¿Cuál?

- a) señal de reloj (CLK)
- b) instrucción actual (bits del registro IR)
- c) estado de la unidad de proceso (flags Z, S, C, O...)
- > d) contador de programa (bits del registro PC)

41. [T3.1] ¿Cuál de las siguientes funciones es una tarea propia de la unidad de control en la CPU?

- > a) decodificar las instrucciones del programa
- b) almacenar instrucciones del programa
- c) realizar operaciones lógicas
- d) almacenar datos del programa

42. En una arquitectura RISC típica:

- a) se usan pocas instrucciones de las disponibles en el conjunto de instrucciones
- > b) suele usarse segmentación
- c) la programación resulta mucho más simple que en una arquitectura CISC
- d) la UC es más compleja que en una arquitectura CISC

43. ¿Cómo actúa el indicador Z del registro de indicadores de estado?

- a) Se pone a 1 cuando el resultado es negativo
- b) Se pone a 0 cuando el resultado es negativo
- > c) Se pone a 1 cuando el resultado de una operación es 0
- d) Se pone a 1 cuando el resultado es positivo

44. Alguna de las siguientes señales no es salida de la unidad de control. ¿Cuál?

- a) señales de lectura y escritura en memoria (RD, WR)
- > b) dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)
- c) códigos de selección en multiplexores, decodificadores, ALU, etc (00, 01, 10, 11...)
- d) señales de carga, habilitación y/o desplazamiento de registros (Load, Enable, ShiftL, ShiftR)

45. Sea un formato de microinstrucción que incluye dos campos independientes de 8 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

- a) 1
- b) 8
- c) 9
- > d) 7

RESPUESTAS TEST SWAD

1 ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?

Elección única Usuario Profesores

- a) Una memoria.
- b) Un contador.
- X c) Un multiplexor.
- d) Un registro.

Puntuación: -0,33

2 [T5.1]

Elección única Alguna de las siguientes NO es una ventaja de la E/S independiente (separada, aislada)

Usuario Profesores

- ✓ • a) Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
- b) Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- c) Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- d) Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)

Puntuación: 1,00

[T5.1FunE/S]

[E14SepTeo09]

3 [P2.2]

Elección única En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor que se indica a continuación, ¿en qué caso ambos programas producen el mismo resultado?

Usuario Profesores

- X a) 0xFFFF FFFF

- 0x0000 001f ffff ffe0 != 0xffff ffff ffff ffe0
- b) 0xAAAA AAAA
 0x0000 0015 5555 5540 != 0xffff fff5 5555 5540
- c) 0x9999 9999
 0x0000 0013 3333 3320 != 0xffff fff3 3333 3320
- 0x1111 1111
- resultado 0x0000 0002 2222 2220
- porque es positivo incluso en complemento a 2
 - d) todos los demás valores se interpretan como negativos, lo primero que hace la suma con signo es extenderlos a 64bit de manera que se activan los 32 bits superiores... resultado radicalmente distinto

Puntuación: -0,33

[P2.2SumSgn]

[E16SepPra06]

Recordar que multiplicar por 32 es desplazar 5 posiciones a la izquierda

- 4 [T3.3]
 Elección ¿Cómo actúa el indicador Z del registro de indicadores de estado?
 única Usuario Profesores
- ✓ • a) Se pone a 1 cuando el resultado es negativo
 b) Se pone a 0 cuando el resultado es negativo
 c) Se pone a 1 cuando el resultado de una operación es 0
 d) Se pone a 1 cuando el resultado es positivo

Puntuación: 1,00

[T3.3CtrlUp]

[E12SepTeo19]

- 5 ¿Es posible utilizar 4 GB de memoria en un sistema cuya CPU emplea E/S mapeada en memoria, cuyo bus de direcciones es de 32 bits y que tiene al menos un puerto de E/S? Supondremos que no se puede emplear ninguna técnica de extensión del bus de direcciones.
 Elección
 única Usuario Profesores
- X a) Sí
 • b) No

- c) Depende de si el número puertos de E/S es muy elevado
- d) Ninguna de las respuestas anteriores es correcta

Puntuación: -0,33

6 Si R0=2, R1=5 y M[3]=3 ¿Qué valor toman R0, R1 y M[3] tras ejecutarse la instrucción XOR 1h[R0],R1?
 Elección única Usuario Profesores

- a) R0=6 , R1=2 , M[3]=5
- b) R0=6 , R1=5 , M[3]=2
- c) R0=2 , R1=5 , M[3]=6
- d) R0=5 , R1=6 , M[3]=2

Puntuación: 0,00

7 [T4.3]
 Elección única En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:
 Usuario Profesores

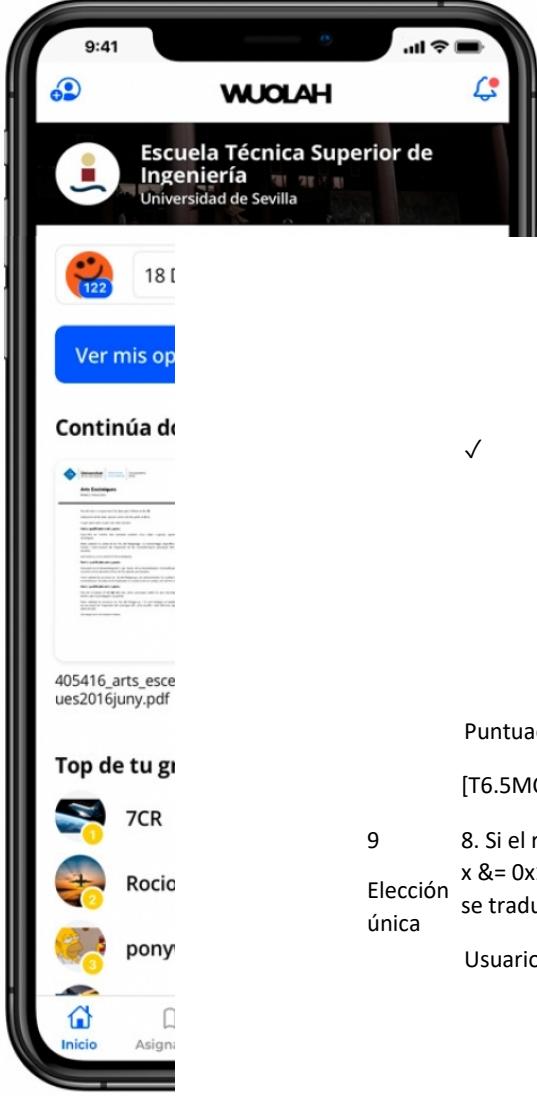
- a) Una disminución de la máxima frecuencia de reloj a la que puede operar el cauce
- b) Una disminución en la posible dependencia de datos
- c) Un mayor retraso en la ejecución de los programas debido al incremento del número de etapas
- ✓ • d) Un incremento de las prestaciones

Puntuación: 1,00

[T4.3Aceler]
 [E13FebTeo30]
 [E17JulTeo14]

8 [T6.5]
 Elección única En los sistemas con una jerarquía de memoria dividida en varios niveles se da el problema de la consistencia o coherencia de los datos entre los distintos niveles. Si una palabra se modifica en un nivel, en algún momento habrá que traspasar ese cambio a los niveles inferiores (más lejanos al procesador). Para ello hay varias políticas:

Usuario Profesores



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

✓ •

- Post-escritura: se retrasa la actualización en los niveles inferiores hasta que el bloque modificado tenga que ser reemplazado
- inferiores hasta que el bloque modificado tenga que ser reemplazado
 - Escritura indirecta: si se modifica una palabra, inmediatamente se modifican los niveles superiores
 - Las respuestas a y b son ciertas
 - Las respuestas a y b son falsas

Puntuación: 1,00

[T6.5MCache]

- 9 8. Si el registro EAX contiene X, la sentencia en C
 $x \&= 0x1;$
 Elección única se traducirá a ensamblador como:

Usuario Profesores

✓ • d) andl \$1, %eax

Puntuación: 1,00

- 10 Respecto a las políticas de arbitraje de buses...

Elección única Usuario Profesores

✓ • d) Todas las repuestas anteriores son ciertas

Puntuación: 1,00

- 11 ¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción mov bx, 5h?

Elección única Usuario Profesores

X a) Direccionamiento relativo a registro base
 b) Direccionamiento inmediato

- c) Direccionamiento implícito
- d) Direccionamiento directo a registro

Puntuación: -0,33

12 [T5.3]
 Elección única ¿A qué tipo de interrupciones pertenecen las condiciones de tiempo real y los fallos hardware?

Usuario Profesores

- a) Enmascarables
- ✓ • b) No enmascarables
- c) Puede ser tanto enmascarables como no enmascarables
- d) Ninguna de las respuestas anteriores es correcta

Puntuación: 1,00

[T5.3ES_IRQ]

13 [T.2.3.1]
 Elección única Para crear espacio en la pila para variables locales sin inicializar suele realizarse la siguiente operación:

Usuario Profesores

- a) Sumar una cantidad positiva a ESP
- ✓ • b) Restar una cantidad positiva a ESP
- c) Sumar una cantidad positiva a EBP
- d) Restar una cantidad positiva a EBP

Puntuación: 1,00

[T2.3.1MarcoP]
 [E16FebTeo12]

14 Respecto al sistema de Entrada / Salida, ¿cuál de las siguientes afirmaciones es incorrecta?
 Elección única Usuario Profesores

- a) La CPU se comunica con el periférico por medio del controlador y de software de E/S.
- b) Un controlador se encarga de la comunicación con la CPU.
- c) Un protocolo sirve para “ponerse de acuerdo” en cosas como velocidad, paridad, nº de bits, etc.

- ✓ • d) La mayoría de los periféricos trabajan a velocidad muy superior a la CPU; por eso es necesario sincronizar.

Puntuación: 1,00

15 En el contexto del lenguaje máquina, el acrónimo ISA suele referirse a:

Elección única Usuario Profesores

- X a) Intel Standard Architecture
b) Industry Standard Architecture
c) Information Security Architecture
• d) Instruction Set Architecture

Puntuación: -0,33

16 [T5.1]

Elección única ¿Cuál de las siguientes funciones no corresponde a un controlador (interfaz) de E/S?
Usuario Profesores

- ✓ • a) Almacenamiento temporal de datos
b) Almacenamiento de programas
c) Comunicación con el dispositivo
d) Comunicación con el microprocesador

Puntuación: 1,00

[T5.1FunE/S]

[E12FebTeo21]

[E12SepTeo12]

17 [T5.1]

Respecto a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:
Elección única Usuario Profesores

- a) El interfaz contiene las controladoras necesarias para conectar los puertos con el procesador
b) El puerto, o interfaz, contiene los controladores necesarios para comunicar el dispositivo con el procesador
c) Cada puerto o interfaz es una línea de comunicación con el procesador. El conjunto de ellos forma el controlador.
✓ • d) La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo

Puntuación: 1,00

[T5.1FunE/S]

[E17FebTeo19]

18 [T2.2.2]

Elección única La extensión de signo a m bits de un número original N de n bits, con m > n, consiste en:

Usuario Profesores

- a) Incrementar la cantidad de bits a m rellenando con unos por la izquierda.
- b) Incrementar la cantidad de bits a m preservando el signo y el valor del número.
- c) Realizar la operación $2^m - N - 1$
- d) Realizar la operación $2^m - N$

Puntuación: 1,00

[T2.2.2OpArit]

[P2Apendice2]

[E17JulTeo02]

19 ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es falsa?

Elección única Usuario Profesores

- a) Ninguna de las afirmaciones es falsa

- b) Es una bifurcación normalmente externa al programa en ejecución

- c) Su objetivo es reclamar la atención del procesador

- d) Solicita que se ejecute un programa específico para tratarla

Puntuación: 1,00

20 La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...

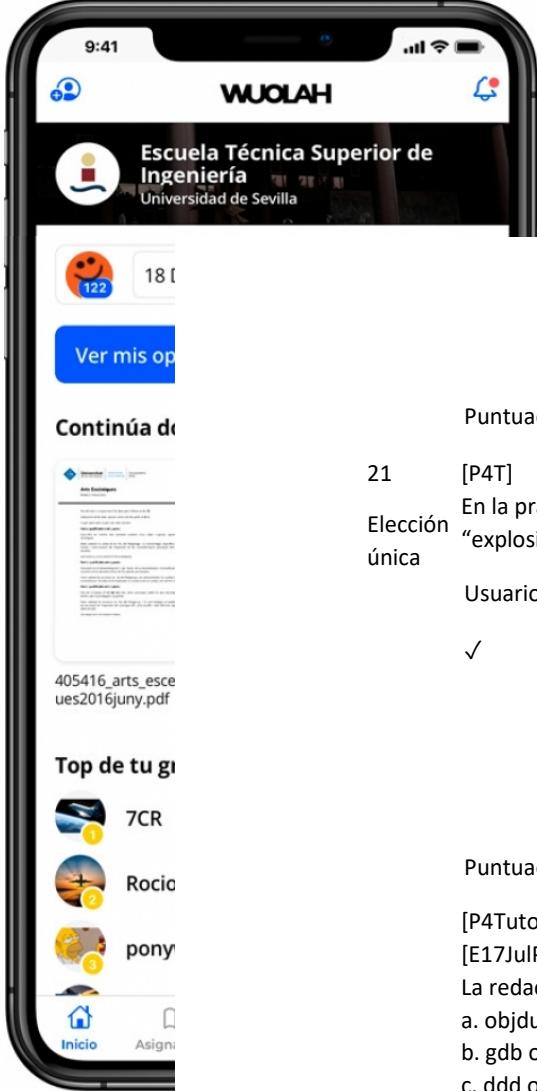
Elección única Usuario Profesores

- a) dos multiplexores de 2 a 1

- b) dos buffers triestado

- c) tres conexiones directas al bus común

- d) tres demultiplexores



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Continúa d...

Puntuación: -0,33

21 [P4T]

Elección única En la práctica de la bomba, el segundo ejercicio consistía en crear un ejecutable sin "explosiones", para lo cual se puede utilizar... (marcar la opción *falsa*)

Usuario Profesores

a) objdump

b) gdb

c) ddd

d) hexedit

405416_arts_escue2016juny.pdf

Top de tu gr...

Puntuación: 1,00

[P4Tutorial]

[E17JulPra16]

La redacción original era: ...para lo cual se puede utilizar...

a. objdump o gdb

b. gdb o ddd

c. ddd o hexedit

d. hexedit u objdump

Con esa redacción, hubo que dar por válidas b. y c. Con depuradores se usaría el método explicado en P4T ejercicio 2, y con hexedit el método explicado en ejercicio 3. Es cierto que hexedit a secas (sin objdump ni depurador) no permite saber dónde hay que modificar nada, pero todo el mundo dice (correctamente) que con hexedit se puede crear el ejecutable sin explosiones, así que esta redacción es menos problemática.

22 [T6.2]

Elección única ¿Cuántas líneas de dirección (patillas) son necesarias para direccionar un chip de memoria DRAM de 4096 x 4?

Usuario Profesores

a) 12

b) 11

c) 10

d) 6

Puntuación: -0,33

[T6.2RAMROM]

[E14SepTeo28]

- 23 [T2.1.2]
 Elección Un archivo .o que contiene código objeto:
 única Usuario Profesores
- a) Puede ejecutarse directamente.
 - b) Incluye el código de las funciones de biblioteca a las que llame.
 - c) Contiene las direcciones definitivas de las variables globales.
 - d) Contiene instrucciones máquina binarias.

Puntuación: 1,00

[T2.1.2Lngjes]
 [P2Tutorial]

- 24 [T5.1]
 Elección Parecidos y diferencias entre los métodos de E/S (señalar la opción incorrecta)
 única Usuario Profesores
- a) sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
 - b) sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo
 - c) no cubre el caso de E/S programada para dispositivos "sin estado" (ej: display 7 segmentos en las diapositivas)
 - d) se suele avisar a la CPU (con una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ como en E/S por DMA
 - e) cubre los casos de IRQ de dispositivo e IRQ de DMA
 - f) la consulta del estado del dispositivo por parte de la CPU se suele/puede incluir en E/S programada y en E/S por IRQ
 - g) originalmente redactado "se suele/puede hacer con E/S programada y con E/S IRQ" - redacción actual más clara, cubre los casos de bucle de espera ocupada y varios dispositivos conectados a la misma IRQ

Puntuación: 1,00

[T5.1FunE/S]
 [E15SepTeo22]

- 25 [T6.3]
 ¿Qué conjunto de componentes permite construir una memoria 256Mx32? (sin que sobren componentes)

Elección Usuario Profesores
única

- a) 16 chips 64Mx4
- b) 32 chips 64Mx4
- c) 16 chips 64Mx16
- d) Ninguna de las anteriores

Puntuación: 1,00

[T6.3Diseño]

[E14FebTeo26]

26 Una instrucción de salto si igual tiene que comprobar el valor de:

Elección Usuario Profesores
única

- a) los bits de signo y desbordamiento
- b) el bit de signo
- c) el bit de cero
- d) el bit de acarreo

Puntuación: 1,00

27 [T6.2]

Elección Usuario Profesores
única Una SRAM de 32Kx8bit (256Kbit) puede venir organizada en 512 filas, dedicando por tanto al decodificador de columnas...

Usuario Profesores

- a) 9 bits
- b) 8 bits
- c) 7 bits
- d) 6 bits

Puntuación: -0,33

[T6.2RAMROM]

[E13SepTeo014]

28 [T4.3]

Elección Usuario Profesores
única Un sistema no segmentado tarda 10 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 5 segmentos con un ciclo de reloj de 4 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:

Usuario Profesores

- a) 2,5
- b) 50
- c) 4
- X d) 5

Puntuación: -0,33

[T4.3Aceler]

- 29 [T5.1]
 Elección Señale cuál de las siguientes opciones es una técnica habitual para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

Única Usuario Profesores

- a) Acceso indirecto a memoria (IMA)
- b) Acceso directo a memoria (DMA)
- X c) E/S por flanco
- d) E/S por nivel

Puntuación: -0,33

[T5.1FunE/S]

[E13FebTeo17]

- 30 [P4T]
 Elección En una bomba como las estudiadas en prácticas, del tipo...

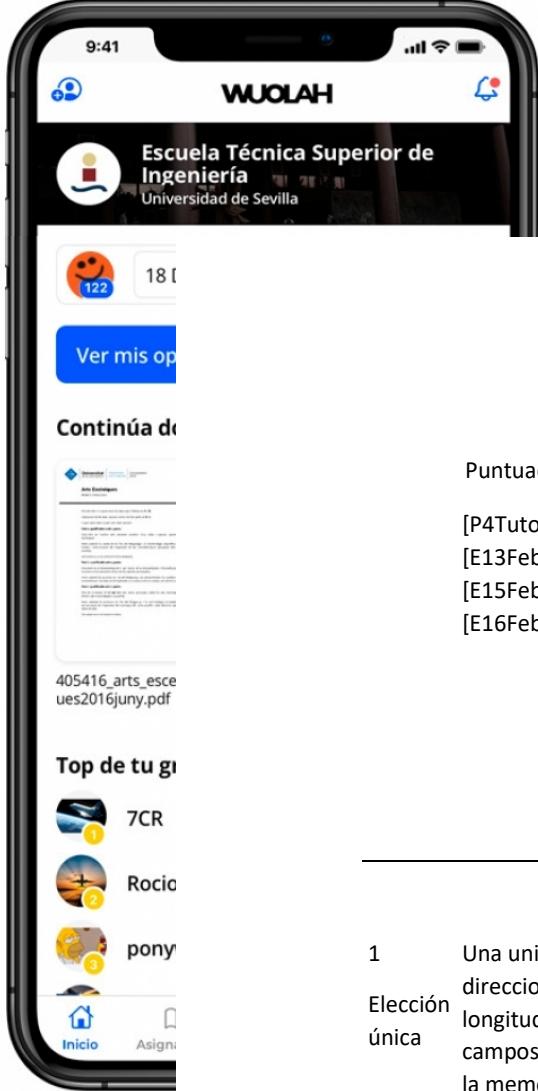
Única

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

Usuario Profesores

- a) el entero 0x804a044
- ✓ • b) el entero almacenado a partir de la posición de memoria
0x804a044
- c) el string almacenado a partir de la posición de memoria
0x24(%esp)



Available on the
App Store

GET IT ON
Google Play

Continúa d

d) ninguna de las anteriores

Puntuación: 1,00

[P4Tutorial]
[E13FebPra11]
[E15FebPra17]
[E16FebPra14]

405416_arts_esce
ues2016juny.pdf

Top de tu gr

7CR
Rocio
pony
Inicio Asign

- 1 Una unidad de control microprogramada con direccionamiento explícito con dos direcciones por microinstrucción, tiene una memoria de control con 35 bits de longitud de palabra. Si las microinstrucciones emplean 15 bits en total para los campos de control y de tipo y condición de salto, el número máximo de palabras de la memoria de control de esta unidad de control microprogramada es de:

Usuario Profesores

- a) 10
- b) 2^{10}
- X c) 2^{20}
- d) 20

Puntuación: -0,33

- 2 Con 8 circuitos de memoria RAM de 1K x 8 se puede crear un memoria de:

Elección Usuario Profesores
única

- a) 1K x 64
- b) 8K x 8
- c) 2K x 32
- ✓ • d) Todas las combinaciones anteriores son posibles

Puntuación: 1,00

- 3 [P4T]
En una bomba como las estudiadas en prácticas, del tipo...
Elección única
0x080486e8 <main+120>: call 0x8048524 <strcmp>

```
0x080486ed <main+125>: test %eax,%eax  
0x080486ef <main+127>: je 0x80486f6 <main+134>  
0x080486f1 <main+129>: call 0x8048604 <boom>  
0x080486f6 <main+134>:
```

...la contraseña es...

Usuario Profesores

- a) el valor que tenga %eax
- b) el string almacenado a partir de 0x80486f6
- X c) el entero almacenado a partir de donde apunta %eax
- d) ninguna de las anteriores

Puntuación: -0,33

[P4Tutorial]

[E14FebPra12]

4 [T2.2.1]

Elección única Si %edx contiene 0xf000 y %ecx contiene 0x0100, el direccionamiento 0x80(%ecx,%edx,2) se refiere a la posición

Usuario Profesores

- a) 0xf182
- b) 0xf280
- c) 0x1e180
- d) Ninguna de las respuestas anteriores es correcta

Puntuación: 0,00

[T2.2.1ModDir]

[E15SepPra03]

5 ¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop:

```
push #1  
push #2  
push #3  
pop a  
push #4  
pop a  
pop a
```

Usuario Profesores

- a) 1, 2, 3 y 4
- ✓ • b) 1
- c) 10
- d) 1 y 2

Puntuación: 1,00

6 [T2.4.1]
 Elección ¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una única función long int al terminar ésta?

Usuario Profesores

- a) La instrucción RET lo almacena en un registro especial de retorno.
- X b) Por convención se guarda en %eax.
- c) Se almacena en pila justo encima de los argumentos de la función.
- d) Ninguna de esas formas es la correcta.

Puntuación: -0,33

[T2.1.4x86-64]
 [T2.2.3CodCon]
 [T2.4.1x86-64]
 [P3Tutorial]
 [E12SepTeo02]
 [E17JulTeo05]

7 [T4.4]
 Elección La predicción de saltos está relacionada con...
 única Usuario Profesores

- a) Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)
- b) Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
- c) Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
- ✓ • d) Los riesgos de control (intenta determinar de antemano el flujo de control)

Puntuación: 1,00

[T4.4Riesgs]
[E15FebTeo13]

8 [T2.1.2]

Elección ¿Cuál de las siguientes afirmaciones sobre la instrucción leave es cierta?
única Usuario Profesores

- a) No es obligatorio usarla. En su lugar puede realizarse una secuencia explícita de operaciones mov y pop

Se ejecuta justo después de retornar de un procedimiento
- b) Lo típico después de retornar (de una función C) es pop/add \$n,%esp y/o mov %eax,...

Equivale a pop %ebp seguida de mov %ebp,%esp
- c) en realidad es mov %ebp,%esp; pop %ebp

Equivale a mov %esp,%ebp seguida de pop %ebp
- d) mov al revés

Puntuación: 0,00

[T2.1.2Lngjes]
[T2.3.1MarcoP]
[E16FebTeo11]

9 [T2.2.3]

Elección En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?
única

Usuario Profesores

- ```
 mov $0, %edi
```

a) 

```
sub $1, %edi
```

  
sub afecta
- ```
        sub %edi, %edi
```

b)

```
adc $0xFFFFFFFF, %edi
```


adc afecta
- c)

```
        mov $-1, %edi
```


mov no afecta a los flags
- ```
 mov $-1, %edi
```

d) 

```
add $0, %edi
```

  
add afecta



13        El tamaño del registro indicador de una memoria asociativa de  $n$  palabras x  $m$  bits/palabra es:  
Elección única    Usuario Profesores

- a)  $n/m$  bits
- X              b)  $m$  bits
- c)  $n$  bits
- d) un bit

Puntuación: -0,33

14        [T3.1]  
Elección única    ¿Cuál de las siguientes funciones no corresponde a la unidad de control de un procesador?  
Usuario Profesores

- ✓              •              a) Calculo de operaciones de coma flotante
- b) Decodificación de las instrucciones
- c) Secuenciamiento de las instrucciones
- d) Generación de las señales de control que provocan la ejecución de cada instrucción

Puntuación: 1,00

[T3.1CamDat]  
[E14FebTeo13]

15        [T2.4.1]  
Elección única    ¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso? (salva-invocante, invocado, etc)  
Usuario Profesores

- a) CL, DX, R8d, R9
- b) RAX, RBX, RCX, RDX
- c) RBX, RSI, RDI
- d) RSP, RBP

Puntuación: 0,00

[T2.4.1x86-64]  
[E13SepTeo09]

16 ¿Cuántas líneas de dirección son necesarias en un memoria RAM de 64 K palabras  
dinámica? ¿Y estática?  
Elección única Usuario Profesores

- a) 8 / 8  
b) 16 / 8  
X c) 16 / 16  
• d) 8 / 16

Puntuación: -0,33

17 [P5.1]

Abajo se ofrece el listado de una función para multiplicar matrices  $C = A \times B$ .

Elección única

```
void mult_matr(float A[N][N], float B[N][N], float C[N][N]){
 /* Se asume valor inicial C = {0,0...} */
 int i,j,k;
 for (i=0; i<N; i++)
 for (j=0; j<N; j++)
 for (k=0; k<N; k++)
 C[i][j] += A[i][k] * B[k][j];
}
```

Suponer que:

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.
- El tamaño de los tipos de datos es como en IA32.
- El compilador optimiza el acceso a  $C[i][j]$  en un registro.

Aproximadamente, ¿qué tasa de fallos se podría esperar de esta función para valores grandes de N?

Usuario Profesores

- a) 1/8  
✓ b) 1/2  
• 1/2 por cada  $B[k][j]$  + 1/32 por cada 16-ésimo  $A[i][k]$   
c) 1/4  
d) 1/16

Puntuación: 1,00

[P5.1Line]  
[E14SepPra09]

[E16FebPra19]

[E17JulPra19]

18 [P5.1]

Elección  
única

Abajo se ofrece el listado de una función para multiplicar matrices  $C = A \times B$ .

```
void mult_matr(float A[N][N], float B[N][N], float C[N][N]){
 /* Se asume valor inicial C = {0,0...} */
 int i,j,k;
 for (i=0; i<N; i++)
 for (j=0; j<N; j++)
 for (k=0; k<N; k++)
 C[i][j] += A[i][k] * B[k][j];
}
```

Suponer que:

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.
- El tamaño de los tipos de datos es como en IA32.
- El compilador optimiza el acceso a  $C[i][j]$  en un registro.

Imaginar que se modifica la última sentencia (el cuerpo anidado) por esta otra

$C[i][j] += A[i][k] * B[j][k];$

de manera que se calcule  $C = A \times B'$  (A por traspuesta de B). Aproximadamente, ¿qué tasa de fallos se podría esperar de esta nueva función para valores grandes de N?

Usuario Profesores

a) 1/2

- b)  $\frac{1}{16}$   
uno de cada 16  $A[i][k]$  y otro de cada 16  $B[i][k]$
- c) 1/8
- d) 1/4

Puntuación: 0,00

[P5.1Line]

[E14SepPra10]

[E16FebPra20]

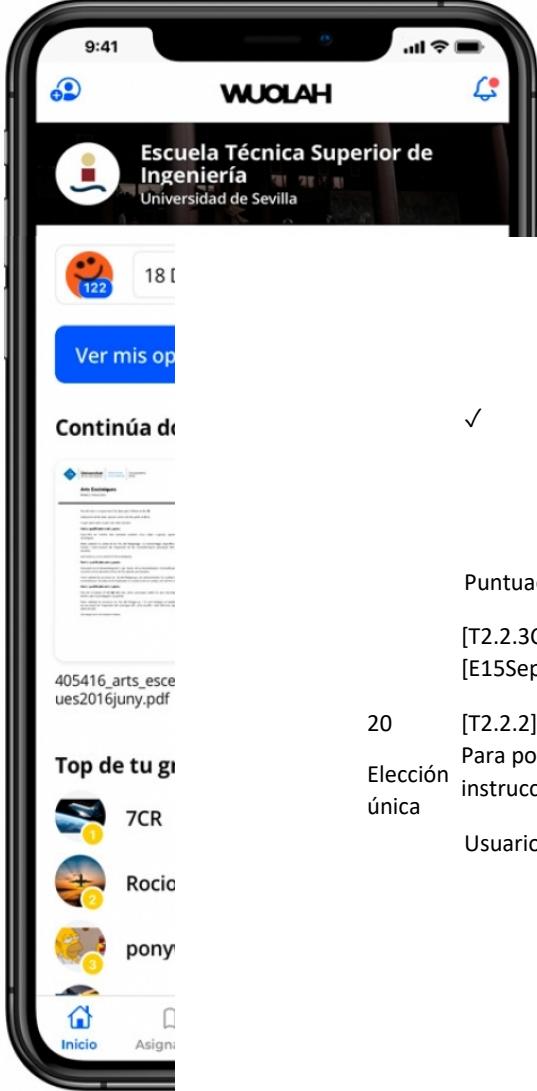
19 [T2.2.3]

Elección  
única

¿Qué combinación de flags aritmético-lógicos corresponde al código de condición b (below)?

Usuario Profesores

a) OF xor SF



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

Continúa d

- ✓ • b) CF

c) OF

d) CF xor OF

Puntuación: 1,00

[T2.2.3CodCon]

[E15SepTeo08]

20 [T2.2.2]

Eleción única Para poner a 1 el bit 5 del registro %edx sin cambiar el resto de bits podemos usar la instrucción máquina:

Usuario Profesores

- a) or \$0x20, %edx
- b) and \$32, %edx
- c) and \$0x5, %edx
- d) or \$0b101, %edx

Puntuación: 0,00

[T2.2.2OpArit]

[E17FebTeo7]

21 En la ejecución de una instrucción...

Eleción única Usuario Profesores

- a) la ALU realiza las operaciones aritméticas y lógicas
- b) la UC activa las señales de control que envía por el bus de direcciones
- c) siempre se altera el registro de estado
- X d) el registro de instrucción se va incrementando para apuntar a la siguiente instrucción

Puntuación: -0,33

22 [T6.1]

Eleción única La tasa de aciertos  $A_{-i}$  del nivel  $i$  de una jerarquía de memoria no depende de:  
(mediante  $_$  se representa subíndice)

Usuario Profesores

- ✓ • a) El ancho de banda  $b_{-i}$  del nivel  $i$ .

Todas las demás se mencionan explícitamente en Tema 6  
tr.24

- b) La capacidad (tamaño)  $s_i$  del nivel  $i$ .
- c) La estrategia de administración de memoria.
- d) La unidad de la transferencia de información  $x_i$  entre el nivel  $i$  y el  $i+1$ .

Puntuación: 1,00

[T6.1ConLoc]

[T6.5MCache]

[E17JulTeo26]

23 ¿Cuál de las siguientes afirmaciones sobre el direccionamiento absoluto es falsa?

Elección Usuario Profesores  
única

- a) El tamaño del operando direccionado queda limitado por el nº de bits del campo dirección de memoria.
- b) La instrucción contiene la dirección de memoria exacta en la que se encuentra el objeto.
- X c) El objeto está en una posición de la memoria principal.
- d) El rango de posiciones direccionables queda limitado por el tamaño del campo de operando.

Puntuación: -0,33

24 [T2.1.3]

Elección ¿Qué valor contendrá el registro edx tras ejecutar las dos instrucciones siguientes?  
única

```
movl $-1, %edx
movb $1, %dl
```

Usuario Profesores

- a) 00000001 00000000 00000000 00000000
- b) 11111111 11111111 11111111 11111111
- c) 11111111 11111111 11111111 00000001
- d) 00000000 00000000 00000000 00000001

Puntuación: 0,00

[T2.1.3ConASM]

[E14SepPra15]

- 25 [T4.3]  
Elección única Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para escribir el resultado en el banco de registros, ¿cuál es la frecuencia máxima del procesador?

Usuario Profesores

- a) 500 MHz
- b) 40 MHz
- c) 66,67 MHz
- d) 200 MHz

Puntuación: 0,00

[T4.3Aceler]

- 26 [T2.2.4]  
Elección única La instrucción JGE / JNL provoca un salto si...  
Usuario Profesores

OF = SF

- a) basta recordar que "Less" no era un flag solo (es OF^SF)  
recordar también que "Below" comprueba CF
- X b) SF = 1  
sería js
- c) CF = 1  
sería jc/jb
- d) SF = 0  
sería jns

Puntuación: -0,33

[T2.2.4SalCon]

[E16FebTeo07]

- 27 En las arquitecturas RISC hay...

Elección única Usuario Profesores

- a) muchos registros y pocos modos de direccionamiento.
- X b) pocas instrucciones muy rápidas con muchos modos de direccionamiento.

- c) pocos modos de direccionamiento y muchos formatos de instrucción.
- d) pocos registros y muchos tipos de instrucciones.

Puntuación: -0,33

28 [T6.5]  
 Elección única ¿En qué se diferencian las estrategias de mantenimiento de coherencia en memoria "escritura directa" y "post-escritura"?

Usuario Profesores

- ✓ •
- a) En cuándo tiene lugar la actualización
  - b) En cómo tiene lugar la actualización
  - c) Tanto en a) como en b)
  - d) Ni en a) ni en b)

Puntuación: 1,00

[T6.1ConLoc]

[T6.5MCache]

29 El 8086 tiene:

Elección única Usuario Profesores

- a) 14 registros de 16 bits más 4 registros de segmento
- b) 16 registros de 16 bits
- c) 16 registros de 20 bits
- d) 14 registros de 16 bits y un bus de direcciones de 20 bits

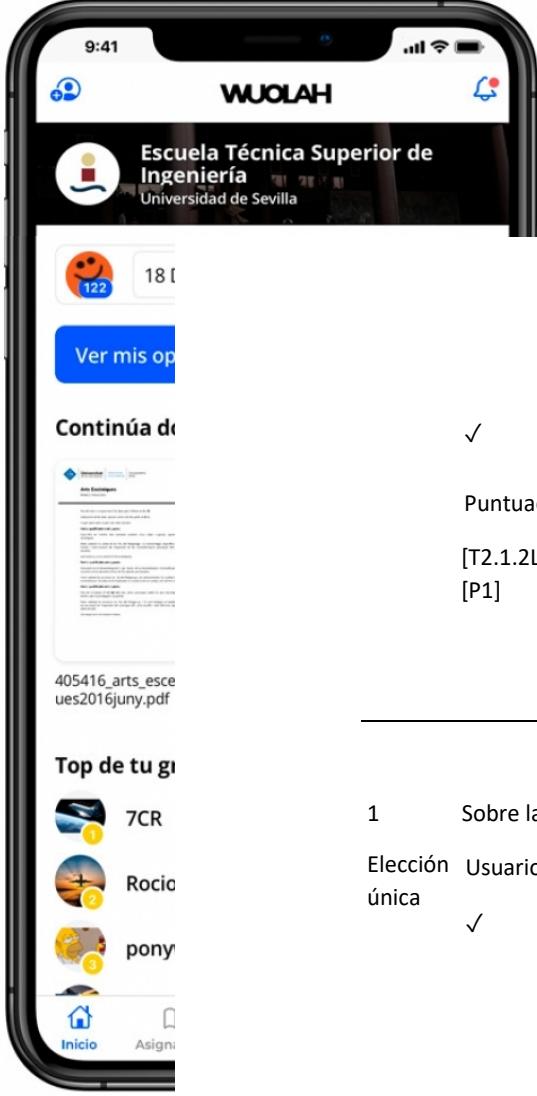
Puntuación: 0,00

30 [T2.1.2]

Elección única ¿Cuál de las siguientes afirmaciones es incorrecta?

Usuario Profesores

- En lenguajes de alto nivel no se tiene acceso a detalles del
- a) procesador como las tablas de páginas, el paso de modo usuario a modo supervisor, el bit de paridad, etc.
  - b) El lenguaje de alto nivel es más portable que el lenguaje máquina.
  - c) En lenguaje ensamblador cada instrucción se corresponde con una instrucción máquina.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

- ✓ • d) En lenguaje ensamblador las instrucciones se escriben en binario.

Puntuación: 1,00

[T2.1.2Lngjes]  
[P1]

405416\_arts\_escuela2016juny.pdf

## Top de tu grupo



- 1 Sobre la E/S mapeada en memoria podemos decir que:

Elección Usuario Profesores única

- ✓ • a) Usa direcciones de memoria para acceder a puertos de E/S  
b) La CPU necesita el pin IO/M#  
c) Dispone de instrucciones especiales de E/S  
d) Todas las respuestas anteriores son falsas

Puntuación: 1,00

- 2 [T2.2.4]

Elección Despues de ejecutar una instrucción de suma sobre dos números con signo de la que sabemos que no provocará overflow (los dos números son pequeños en valor absoluto), queremos comprobar si el resultado de la suma es menor que 0. ¿Qué flag necesita comprobar la instrucción de salto condicional equivalente a... ?

if (resultado<0) then goto label

Usuario Profesores

- a) OF  
b) CF  
c) ZF  
d) SF

Puntuación: 1,00

[T2.2.3CodCon]  
[T2.2.4SalCon]

[E14SepTeo19]

- 3 ¿Qué tipo de localidad de las referencias se define como: "si se referencia un elemento, volverá a ser referenciado pronto"?

Elección Usuario Profesores  
única

- a) Localidad secuencial
- b) Localidad espacial
- c) Localidad iterativa
- ✓      •      d) Localidad temporal

Puntuación: 1,00

4 [T2.2.2]

Elección Si EAX=2, EBX=5 y M[3]=3, ¿qué valores quedan tras ejecutarse la instrucción XOR  
única %BL, 1(%EAX)?

Usuario Profesores

- a) EAX=2 , EBX=5 , M[3]=6
- b) EAX=5 , EBX=6 , M[3]=2
- c) EAX=6 , EBX=2 , M[3]=5
- d) EAX=6 , EBX=5 , M[3]=2

Puntuación: 0,00

[T2.2.1ModDir]

[T2.2.2OpArit]

5 [T2.2.1]

Elección Si EBX=0x00002a00 y EDI=0x0000000a, ¿cuál es la dirección efectiva en la  
única instrucción add 0x64(%ebx,%edi,2),%eax?

Usuario Profesores

- a) 0x00002a78
- b) 0x00002a70
- c) 0x000054dc
- d) 0x00005478

Puntuación: 0,00

[T2.2.1ModDir]

[E15FebPra05]

6 Sobre un sistema que utiliza un esquema de memoria virtual con segmentación  
Elección paginada podemos decir que:  
única

Usuario Profesores

- a) Es un sistema de correspondencia entre direcciones virtuales y direcciones en memoria caché
- b) Entre disco y memoria principal se transfieren segmentos completos
- c) Las respuestas a y b son ciertas
- d) Las respuestas a y b son falsas

Puntuación: 0,00

7 [T2.2.4]

Elección Al ejecutar el fragmento de código:

única

```
leal -4(%eax), %edx
cmpl $9, %edx
ja .L2
```

se salta a .L2 si el contenido del registro %eax:

Usuario Profesores

- a) es mayor o igual que 48
- X b) está dentro del intervalo [48,57]
- c) está fuera del intervalo [48,57]
  - d) es mayor o igual que 57

Puntuación: -0,33

[T2.2.1ModDir]

[T2.2.2OpArit]

[T2.2.3CodCon]

[T2.2.4SalCon]

[E14SepTeo20]

8 Si d es un desplazamiento, r un registro índice e i una constante apropiada, el modo de direccionamiento indexado con postautoincremento realiza...

Elección

única Usuario Profesores

- a)  $r = r + i$  ; dirección efectiva =  $r + d$
- ✓ • b) dirección efectiva =  $r + d$  ;  $r = r + i$
- c)  $r = r - i$  ; dirección efectiva =  $r + d$
- d) dirección efectiva =  $r + i$  ;  $r = r + d$

Puntuación: 1,00

9 ¿Cuál de los siguientes grupos de instrucciones podrá pertenecer a un procesador con E/S mapeada en memoria?  
Elección única Usuario Profesores

- a) IN, LOAD, OUT
- b) IN, LOAD, MOV
- ✓ • c) LOAD, MOV, STORE
- d) Ninguno de los anteriores

Puntuación: 1,00

10 [T2.3.1]  
Elección única Al llamar a una función de 2 argumentos foo(arg1, arg2), ¿cuál es el orden correcto de las operaciones? (suponiendo convención de llamada x86 cdecl, y que foo requiere ajustar marco de pila, esto es, salvar %ebp)

Usuario Profesores

- a) push arg1, push arg2, push %ebp, call foo
- b) push arg1, push arg2, call foo, push %ebp
- X • c) push arg2, push arg1, push %ebp, call foo
- d) push arg2, push arg1, call foo, push %ebp

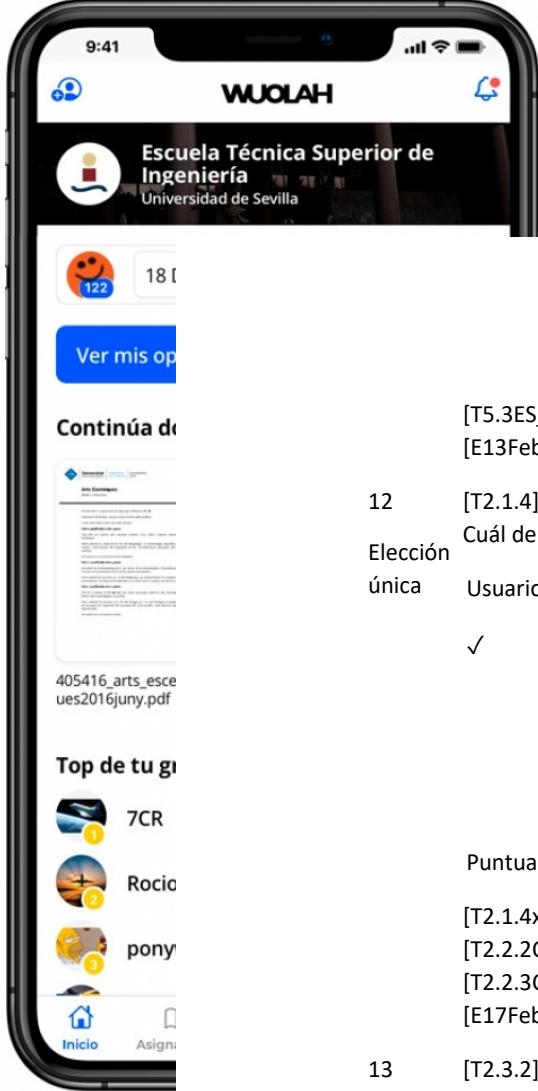
Puntuación: -0,33

[T2.3.1MarcoP]  
[E14SepTeo03]  
[E16FebTeo17]

11 [T5.3]  
Elección única ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es cierta?  
Usuario Profesores

- a) Su objetivo es incrementar el ancho de banda con el dispositivo
- ✓ • b) Es una bifurcación normalmente externa al programa en ejecución
- c) Permite realizar transferencias sin el control de un programa
- d) Sigue la ejecución del programa

Puntuación: 1,00



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## Continúa d...

405416\_arts\_escue  
ues2016juny.pdf

## Top de tu gr...

7CR  
Rocio  
pony  
Inicio Asign.

[T5.3ES\_IRQ]  
[E13FebTeo18]

12 [T2.1.4]  
Elección  
única Usuario Profesores

- ✓ • a) movl %r8, %eax  
b) testl %edx, %edx  
c) addq \$1, %rcx  
d) movl (%rdi,%rcx,4), %edx

Puntuación: 1,00

[T2.1.4x86-64]  
[T2.2.0OpArit]  
[T2.2.3CodCon]  
[E17FebTeo5]

13 [T2.3.2]  
Elección  
única Usuario Profesores

- a) %eax, %edx, %ecx  
✓ • b) %ebx, %esi, %edi  
c) %esi, %edi  
d) %eax, %ebx, %ecx, %edx

Puntuación: 1,00

[T2.3.2Conven]  
[E16FebTeo16]  
El enunciado asume convención cdecl

14 [T2.1.4]  
Elección  
única Usuario Profesores

- a) los punteros (void\*) pasan de 32 a 64 bits  
b) los enteros largos (long) pasan de 32 a 64 bits  
c) long double pasa de 10/12 B a 16 B  
✓ • d) el tipo double pasa de 4 B a 8 B

siempre 8 B

Puntuación: 1,00

[T2.1.4x86-64]

[E16SepTeo17]

15 [T4.3]

Elección En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:

única Usuario Profesores

a) Un mayor retraso en la ejecución de los programas debido al incremento del número de etapas

✓ • b) Un incremento de las prestaciones

c) Una disminución de la máxima frecuencia de reloj a la que puede operar el cauce

d) Una disminución en la posible dependencia de datos

Puntuación: 1,00

[T4.3Aceler]

[E13FebTeo30]

16 [T5.1]

Elección ¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?

única Usuario Profesores

a) En las transferencias síncronas

• b) En las transferencias asíncronas

X c) En ambas

d) En ninguna

Puntuación: -0,33

[T5.1FunE/S]

[E12FebTeo25]

17 ¿Cuál de los siguientes elementos no forma parte del canal de un controlador de acceso directo a memoria?

Elección Usuario Profesores

• a) Registro de prioridades.

b) Registro contador.

X c) Todos los elementos de las otras respuestas forman parte de un canal de DMA.

d) Registro de dirección.

Puntuación: -0,33

- 18 Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?

Elección única Usuario Profesores

a)  $k + n \cdot 2^m$

X b)  $k + n^m$

• c)  $k + n \cdot (2^m - 1)$

d) Ninguno de los anteriores

Puntuación: -0,33

- 19 El tamaño del registro indicador de una memoria asociativa de n palabras x m bits/palabra es:

Elección única Usuario Profesores

• a) n bits

b) m bits

X c)  $n/m$  bits

d) un bit

Puntuación: -0,33

- 20 ¿En qué técnica para determinar la dirección de comienzo de la rutina de servicio de interrupción se fija dicha dirección en los circuitos de la CPU?

Elección única Usuario Profesores

a) Direccionamiento absoluto.

• b) Direcciones fijas.

c) Envío de instrucción de bifurcación completa.

X d) Direccionamiento relativo.

Puntuación: -0,33

- 21 La convención de llamada Linux/GCC IA32 considera, respecto a convenios de uso de registros:
- Elección única Usuario Profesores
- a) 8 registros salva-invocante, 6 registros salva-invocado, y 2 especiales
  - b) 3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales
  - X c) Algunos registros para pasar argumentos, otros salva-invocante, otros salva-invocado, dos especiales
  - ✓ d) Algunos registros salva-invocante, otros salva-invocado, uno especial

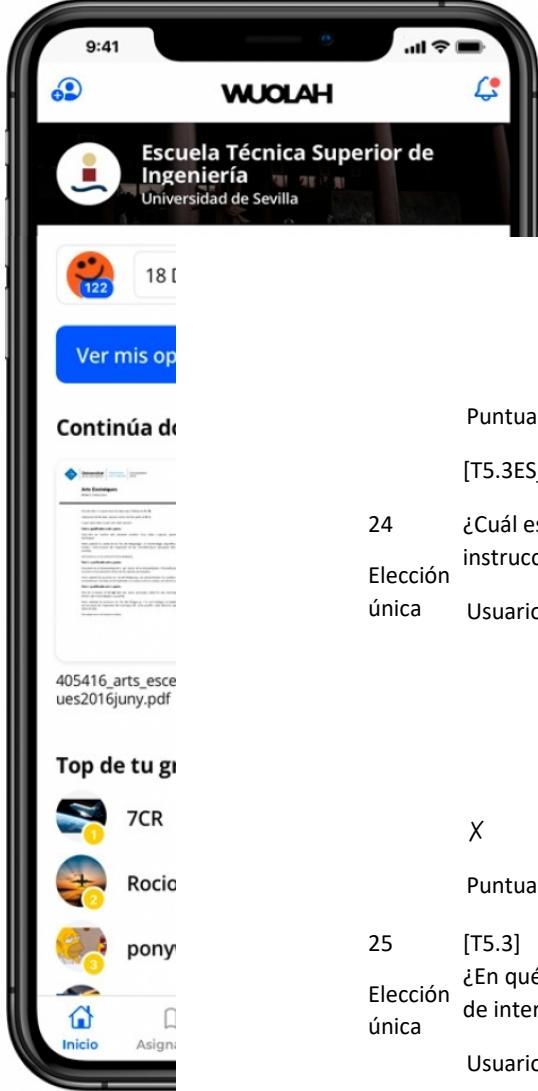
Puntuación: -0,33

- 22 [T6.1]
- Elección única Si el tiempo de acceso a la memoria cache es de 2 ns y el tiempo necesario para tratar un fallo de cache es de 80 ns, ¿cuál es la tasa de aciertos necesaria para que el tiempo medio de acceso al sistema de memoria sea de 10 ns?
- Usuario Profesores
- a) 0.8
  - b) 0.75
  - c) 0.95
  - ✓ d) 0.9

Puntuación: 1,00

[T6.1ConLoc]  
[T6.5MCache]  
[E14FebTeo16]

- 23 [T5.3]
- Elección única La técnica de sondeo, escrutinio o "polling"...
- Usuario Profesores
- a) Se utiliza para identificar el destino de una interrupción
  - b) Permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos
  - X c) En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción
  - d) Todas las respuestas anteriores son falsas



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## Continúa d

Puntuación: -0,33

[T5.3ES IRQ]

- 24 ¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila: push #4; push #7; add; push #10; sub?  
Elección única Usuario Profesores

a) 4, 11, 1

- b) 1

c) 11, 1

d) 4, 7, 10

405416\_arts\_esce ues2016juny.pdf

## Top de tu grupo

7CR

Rocio

pony

Inicio

Asign

Puntuación: -0,33

- 25 [T5.3]  
¿En qué método para determinar la dirección de comienzo de una rutina de servicio de interrupción se envia parte de dicha dirección?  
Elección única Usuario Profesores

a) Direccionamiento absoluto

b) Direccionamiento mediante instrucción de bifurcación

- c) Direccionamiento relativo

d) Direccionamiento indirecto

Puntuación: -0,33

[T5.3ES IRQ]

- 26 [T6.2]  
La memoria DRAM:  
Elección única Usuario Profesores

X

a) Es menos densa que la memoria SRAM

Se denomina dinámica porque para mantener almacenado un

- b) dato hay que recargarlo cada cierto tiempo en un ciclo de refresco

c) Necesita 6 transistores por cada celda

d) Se inventó en la década de los 90

Puntuación: -0,33

[T6.2RAMROM]  
[E13SepTeo29]

27 La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...  
Elección única Usuario Profesores

- a) tres conexiones directas al bus común
- b) tres demultiplexores
- ✓ • c) dos multiplexores de 2 a 1
- d) dos buffers triestado

Puntuación: 1,00

28 [T2.2.3]  
Un overflow nunca puede ocurrir cuando:  
Elección única Usuario Profesores

- a) se suman dos números negativos  
0x80000000 + 0xffffffff
- b) se suman dos números positivos  
0xfffffff + 0x1
- c) se resta un número positivo de un número negativo  
0x80000000 - 0x00000001
- ✓ • d) se suma un número positivo a un número negativo  
resultado queda entre ambos => se puede representar

Puntuación: 1,00

[T2.2.3CodCon]  
[E16FebTeo08]

29 [T2.4.1]  
Comparando las convenciones de llamada de gcc Linux IA32 con x86-64 respecto a registros  
Elección única Usuario Profesores

- En IA32 %ecx es salva-invocante, y en x86-64 %rcx es salva-invocante también
- a) %rcx es 4º argumento - salva-invocante, ok

- En IA32 %ebx es salva-invocante, pero en x86-64 %rbx es
- b) salva-invocado  
 %ebx es salva-invocado (verde en tr.2.3.36)
- En IA32 %ebp es especial (marco de pila), y en x86-64 %rbp
- c) también  
 %rbp es salva-invocado (verde en tr.2.4.5)
- En IA32 %esi es salva-invocado, y en x86-64 %rsi es salva-
- d) invocado también  
 %rsi es 2º argumento - salva-invocante (amarillo en tr.2.4.5)

Puntuación: 0,00

[T2.4.1x86-64]  
 [E15SepPra11]  
 [E17JulTeo06]

- 30      [T6.5]
- Elección    En las políticas anticipativas de extracción de cache, ¿cuál de ellas se caracteriza por preextraer el bloque  $i+1$  si se referencia al bloque  $i$  y se produce falta de bloque?
- única
- Usuario Profesores

- a) Preextracción indexada
  - b) Preextracción marcada
  - c) Preextracción por falta
  - X       d) Preextracción siempre
- 

- 1       Respecto a las unidades de control nanoprogramadas:

- Elección    Usuario Profesores
- única
- X
- a) La realización nanoprogramada de una unidad de control es más rápida que la micropogramada.
  - b) El diseño de las unidades de control nanoprogramadas debe ser vertical.
  - c) Suponiendo una memoria de microprograma con  $n$  microinstrucciones de  $w$  bits cada una, de las cuales  $2^m$  son distintas, el ahorro en bits si se utiliza nanoprogramación es  $(n \cdot m + 2^m \cdot w) - n \cdot w$ .

- La anchura de la memoria de nanoprograma es la misma que d) la de memoria de microprograma en un diseño de la misma unidad de control que no usara nanoprogramación.

Puntuación: -0,33

2 ¿Cuántos bytes puede transmitir como máximo el controlador de acceso directo a memoria 8237 de forma consecutiva?  
 Elección única Usuario Profesores

- a) 128 KB
- b) Existe al menos un modo de funcionamiento sin límite máximo
- c) 32 KB
- X d) 64 KB

Puntuación: -0,33

3 [T2.2.3]  
 Elección única ¿Cuál de las siguientes parejas de mnemotécnicos de ensamblador IA32 corresponden a la misma instrucción máquina?  
 Usuario Profesores

- ✓ • a) JZ (saltar si cero), JE (saltar si igual)
- b) SAR (desplazamiento aritmético a la derecha) / SHR (desplazamiento lógico a la derecha)
- c) CMP (comparar), SUB (restar)
- d) JC (saltar si acarreo), JL (saltar si menor, para números con signo)

Puntuación: 1,00

[T2.2.2OpArit]  
 [T2.2.3CodCon]  
 [T2.2.4SalCon]  
 [E12FebTeo15]

4 [T3.3CtrlUp]  
 Elección única En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de

Usuario Profesores



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

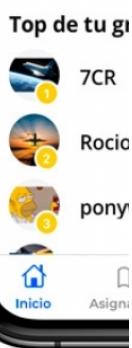
## Continúa d



- X                    a) 16 bits  
        •              b) 17 bits  
                       c) 4 bits  
                       d) 5 bits

Puntuación: -0,33

[T3.3CtrlUp]



- 5                    [P3.1]  
                       Elección              ¿Cuál es el popcount (peso Hamming, nº de bits activados) del número 19?  
                       única                Usuario Profesores

- ✓                    a) 2  
        •              b) 3  
                       19 = 0x13 = 16+3 = 0b0001 0011 -> popcount 3  
                       c) 4  
                       d) 5

Puntuación: 1,00

[P3.1Pcount]  
 [T2.2.5BWhile]  
 [E17FebPra10]

- 6                    [T5.4]  
                       Elección              Si el acceso directo a memoria se realiza mediante robo de ciclo:  
                       única                Usuario Profesores

- ✓                    a) es posible que la ejecución de una operación elemental de transferencia en el bus sea temporalmente detenida  
        •              b) es posible que la ejecución de una instrucción máquina sea temporalmente detenida  
                       c) es necesario que termine de ejecutarse la instrucción máquina actual para comenzar una transferencia por DMA  
                       d) ninguna de las anteriores es cierta

Puntuación: 1,00

[T5.4ES\_DMA]

- 7                    Una memoria SRAM tiene una capacidad de 64 Kbits y precisa 12 líneas de dirección para su manejo. Indique cuál es el tamaño de palabra de dicha memoria:

Elección Usuario Profesores  
única

- ✓ • a) 16 bits  
b) 8 bits  
c) 64 bits  
d) 32 bits

Puntuación: 1,00

8 [T6.2]

Elección ¿Cuál de las siguientes afirmaciones es falsa?  
única Usuario Profesores

- a) La lectura de un bit de la matriz de almacenamiento de una memoria DRAM proporciona una señal mucho más débil que la suministrada por los inversores de una celda de memoria SRAM.
- b) Las memorias DRAM son en general mucho más rápidas que las SRAM
- c) Las memorias DRAM presentan generalmente una capacidad de almacenamiento mucho mayor que las SRAM.
- d) Una celda DRAM sólo necesita un transistor y un condensador.

Puntuación: 1,00

[T6.2RAMROM]

9 [T2.1.2]

Elección En Linux IA32, si gcc usa la instrucción leave se puede asegurar que en ese punto del  
única programa

Usuario Profesores

ya no hay variables locales que destruir

- a) puede usarse leave para destruir rápidamente todas las variables locales, siempre que no haya que recuperar registros salva-invocado
- ya no se hacen llamadas anidadas y por tanto no hay parámetros que ocupen espacio en pila
- b) puede usarse leave habiendo espacio reservado en pila para argumentos de llamadas anidadas, siempre que no haya que recuperar registros salva-invocado

- X                    correspondería emitir la secuencia de salida pop/ret, pero
- c) leave hace lo mismo y ocupa menos espacio
  - leave no equivale a ret, y sólo hace pop %ebp
  - d) ya no hay registros salva-invocado que recuperar
  - tal vez porque no había ninguno, para empezar

Puntuación: -0,33

[T2.1.2Lngjes]

[T2.3.1MarcoP]

[E16SepTeo13]

- 10                  [T2.2.1]  
 Elección            ¿Cuál de las siguientes instrucciones convierte `%eax = 5 * %eax`?  
 única                1) `mov (%eax, %eax, 4), %eax`  
                        2) `lea (%eax, %eax, 4), %eax`

Usuario Profesores

- a) Sólo 2
- b) Ambas 1 y 2
- c) Ninguna

X                    d) Sólo 1

Puntuación: -0,33

[T2.1.3ConASM]

[T2.2.1ModDir]

[E14FebTeo02]

- 11                  Elección            Los valores de los registros Argumento y Máscara de una memoria asociativa son los  
                       única                siguientes:  
                       Argumento: 010010  
                       Máscara: 101011  
                       Si en la primera posición de la memoria está almacenado el valor 0, y las siguientes  
                       celdas de memoria tienen el valor de la celda inmediatamente anterior  
                       incrementado en 1, siendo el valor de la última celda el 7, ¿cuál sería el valor del  
                       registro indicador o de marca?

Usuario Profesores

- a) 00100010
- b) 01001011
- c) 10101100
- d) Ninguno de los anteriores

Puntuación: 0,00

12 Con 8 circuitos de memoria RAM de 1K x 8 se puede crear un memoria de:

Elección Usuario Profesores  
única

a) 1K x 64

b) 8K x 8

c) 2K x 32

✓ • d) Todas las combinaciones anteriores son posibles

Puntuación: 1,00

13 ¿Cuál de las siguientes afirmaciones sobre los lenguajes ensambladores es falsa?

Elección Usuario Profesores  
única

✓ • a) A cada sentencia le corresponde un conjunto preestablecido de instrucciones máquina.

b) A cada sentencia le corresponde una única instrucción de lenguaje máquina.

c) Usan símbolos nemotécnicos para los códigos de operación de las instrucciones.

d) Usan etiquetas para las direcciones de memoria.

Puntuación: 1,00

14 [T2.3.1]

Elección Si el contenido de ESP es 0xAC00 antes de ejecutar la instrucción push %ebx. ¿Cuál será su contenido tras ejecutarla?

Única Usuario Profesores

a) 0xAC02

b) 0xAC04

✓ • c) 0xABFC

d) 0xABFE

Puntuación: 1,00

[T2.3.1MarcoP]

[E15FebPra08]

15 [T2.1.4]

En un sistema IA32 Linux, ¿cuál es el tamaño de un long?



**Descarga la APP de Wuolah.**  
Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

Continúa d'

Elección única Usuario Profesores

a) 6 bytes

X b) 8 bytes

c) 2 bytes

• d) 4 bytes

Puntuación: -0,33

[T2.1.4x86-64]

[E14FebTeo06]

16 [T2.2.1]

Elección única En el direccionamiento indirecto a través de registro, la dirección efectiva...

Usuario Profesores

• a) se encuentra en un registro general del procesador.

b) se encuentra en el registro de instrucción.

c) se encuentra en una dirección de memoria.

d) se calcula como la suma del contenido de dos registros.

Puntuación: -0,33

[T1.2ConBas]

[T2.2.1ModDir]

17 [T5.1]

Elección única Alguna de las siguientes NO es una técnica de E/S de las estudiadas en clase:

Usuario Profesores

a) E/S controlada por interrupciones

b) E/S mediante Acceso Directo a Memoria

c) E/S programada

✓ • d) E/S asíncrona

Puntuación: 1,00

[T5.1FunE/S]

[E14SepTeo10]

18 [T2.1.4]

Elección única La arquitectura x86-64 tiene:

Usuario Profesores

- a) 32 registros RPG de 64 bits
  - b) 64 registros RPG de 64 bits
  - c) 8 registros de propósito general (RPG) de 64 bits (%rax, %rbx, ... %rsp, %rbp)
  - ✓ d) 16 registros RPG de 64 bits

Puntuación: 1,00

[T2.1.4x86-64]

[E13SepTeo08]

19 ¿En qué tipo de política de arbitraje de buses se desperdicia ancho de banda cuando alguno de los posibles maestros no tiene nada que transmitir?  
Elección única Usuario Profesores

- - a) Política dinámica FIFO
    - b) Política estática
    - c) Política dinámica de prioridad fija
    - d) Ninguna de las respuestas anteriores es cierta

Puntuación: 0,00

20 [T3.1] ¿Cuál de las siguientes funciones no corresponde a la unidad de control de un procesador?

Elección única

Usuario Profesores

- ✓     • a) Generación de las señales de control que provean la ejecución de cada instrucción

      b) Calculo de operaciones de coma flotante

      c) Decodificación de las instrucciones

      d) Secuenciamiento de las instrucciones

Puntuación: 1.00

[T3.1CamDat]

[F14FebTeo13]

21 [T2.1.2] Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:

Elección única

Usuario Profesores

- ✓ • a) compilador  
b) ensamblador  
c) intérprete  
d) simulador

Puntuación: 1,00

[T2.1.2Lngjes]

[E14SepTeo14]

22 Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

Elección única Usuario Profesores

- a) esi  
b) edi  
✓ • c) eax  
d) ebx

Puntuación: 1,00

23 [T2.1.2]

Elección única ¿Cómo se devuelve en ensamblador x86 Linux gcc el valor de retorno de una función al terminar ésta?

Usuario Profesores

- a) Se almacena en pila justo encima del (%ebp) del invocado  
b) Se almacena en pila justo encima de los argumentos de la función  
c) La instrucción RET lo almacena en un registro especial de retorno  
✓ • d) Por convención se guarda en %eax

Puntuación: 1,00

[T2.1.2Lngjes]

[P3Tutorial]

[E12FebTeo02]

24 ¿Cuál de las siguientes afirmaciones acerca de un sistema con un bus único es falsa?

Elección única Usuario Profesores

- X            a) El procesador y los dispositivos pueden funcionar a diferentes velocidades si funciona de forma asíncrona
- b) El procesador y los dispositivos pueden funcionar a diferentes velocidades si funciona de forma síncrona
- c) Sólo un dispositivo puede escribir en el bus en un instante dado
- d) Se utilizan las mismas líneas de control para conectar todos los dispositivos

Puntuación: -0,33

25        ¿Cuál de las siguientes tareas no es responsabilidad de un circuito de interfaz o controlador de periféricos sencillo?

Elección única    Usuario Profesores

- ✓            a) Recibir señales de control desde el procesador
- b) Ejecutar el programa de transferencia de información entre el procesador y los dispositivos de E/S
- c) Adaptar el formato de las señales
- d) Ajustar la temporización entre el procesador y los dispositivos de E/S

Puntuación: 1,00

26        [P3T]

Elección única    Usuario Profesores

- a) ecx
- b) edx
- ✓            c) edi
- d) esi

Puntuación: 1,00

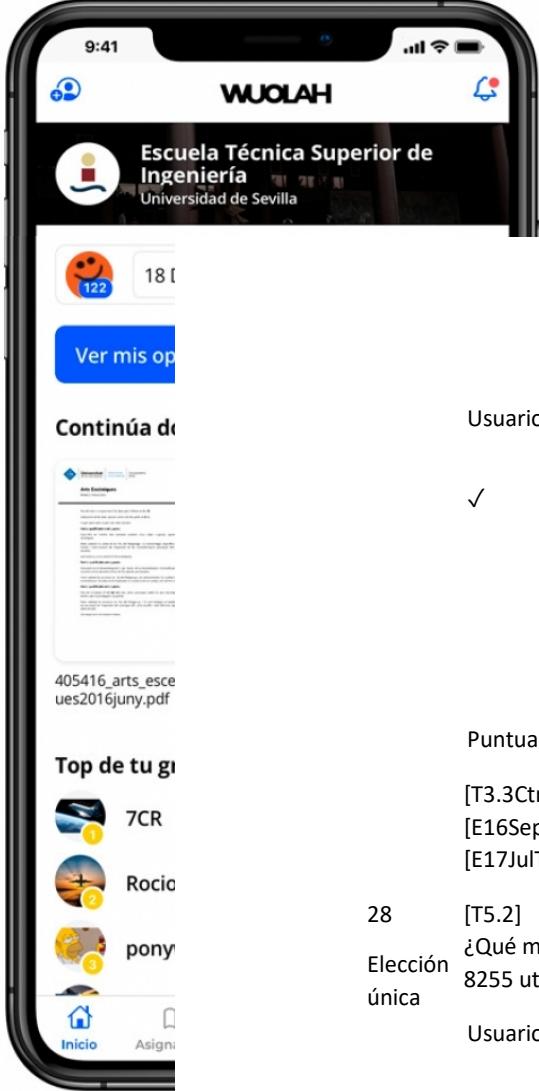
[P3Tutorial]

[T2.4.1x86-64]

[E17JulPra12]

27        [T3.3]

Elección única    Dado un camino de datos concreto, un posible formato de micropogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## Continúa d'

### Usuario Profesores

- ✓ • microbifurcaciones  
a) T3 tr.49

- b) longitud relativa de microinstrucción  
c) codificación  
d) solapamiento

Puntuación: 1,00

[T3.3CtrlUp]  
[E16SepTeo23]  
[E17JulTeo11]

28 [T5.2]

Elección única ¿Qué modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?

### Usuario Profesores

- a) 2  
b) 3  
c) 0  
d) 1

Puntuación: 0,00

[T5.2ESProg]

29 Indique cuál de las siguientes características no es cierta en el direccionamiento indirecto a memoria a través de memoria:

### Elección única Usuario Profesores

Permite una gran capacidad de direccionamiento al poderse  
a) utilizar todos los bits de la palabra de memoria como dirección.

- ✓ • b) La instrucción contiene la dirección de memoria exacta en que se encuentra el objeto.  
c) No se requieren cálculos previos para conocer la dirección final.  
d) Son necesarios dos accesos a memoria (aparte de la fase de búsqueda de instrucción) para acceder al objeto.

Puntuación: 1,00

30 El 8086 tiene:

Elección Usuario Profesores  
única

- a) 14 registros de 16 bits y un bus de direcciones de 20 bits
- b) 14 registros de 16 bits más 4 registros de segmento
- X c) 16 registros de 16 bits
- d) 16 registros de 20 bits

Puntuación: -0,33

---

1 [P2T]

¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa en lenguaje C? (el fichero sin extensión es un ejecutable):  
Elección única

Usuario Profesores

- a) fich.c → fich.s → fich → fich.o
- b) fich → fich.s → fich.o → fich.c
- c) fich.c → fich.o → fich.s → fich
- ✓ d) fich.c → fich.s → fich.o → fich

Puntuación: 1,00

[P2Tutorial]

[T2.1.2Lngjes]

[E13FebPra01]

2 [P2A2]

Alguno de los siguientes no es un nombre de registro en una máquina x86-64 en modo 64 bits  
Elección única

Usuario Profesores

sih

- ✓ • a) %sil en todo caso, no %sih  
Ver libro Hallaron Figura 3.35
- b) r12w
- c) r8d
- d) spl

Puntuación: 1,00

[P2Apendice2]  
[T2.4.1x86-64]  
[E16SepPra11]

### 3 [P1]

Sobre el programa ensamblador:

Elección Usuario Profesores  
única

X

- La calidad de un programa ensamblador afectará
- a) menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
  - Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
  - b) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
  - c) d) Todas las respuestas son ciertas.

Puntuación: -0,33

[P1]

[P2Tutorial]

[T2.1.2Lngjes]

### 4 [P4T]

En la práctica de la bomba, el segundo ejercicio consistía en crear un ejecutable sin “explosiones”, para lo cual se puede utilizar... (marcar la opción \*falsa\*)

Elección  
única

Usuario Profesores

✓

•

- a) gdb
- b) ddd
- c) hexedit
- d) objdump

Puntuación: 1,00

[P4Tutorial]

[E17JulPra16]

La redacción original era: ...para lo cual se puede utilizar...

- a. objdump o gdb
- b. gdb o ddd
- c. ddd o hexedit
- d. hexedit u objdump

Con esa redacción, hubo que dar por válidas b. y c. Con depuradores se usaría el método explicado en P4T ejercicio 2, y con hexedit el método explicado en ejercicio 3. Es cierto que hexedit a secas (sin objdump ni depurador) no permite saber dónde hay que modificar nada, pero todo el mundo dice (correctamente) que con hexedit se

puede crear el ejecutable sin explosiones, así que esta redacción es menos problemática.

## 5 [P3.2]

La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la Elección siguiente versión de parity5:

única

```
int parity5(unsigned * array,
 int len){
 int i,j, result = 0;
 unsigned x;
 for(i = 0; i<len; i++){
 x=array[i];
 for(j=1; j<8*sizeof(int); j*=2)
 x ^= x >> j;
 result += x & 0x1;
 }
 return result;
}
```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase, en las condiciones del bucle for interno.

Esta función parity5:

Usuario Profesores

Produce siempre el resultado correcto

- a) Efectivamente, con este for también se cumple que el LSB (bit menos significativo) sigue siendo el XOR lateral de todos los bits.  
b) Fallaría con array={0,1,2,3}  
c) Fallaría con array={1,2,4,8}  
d) No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: 0,00

[P3.2Parity]

[E17FebPra14]

El bucle sugerido por el guión de prácticas hubiera sido:

for (j=8\*sizeof(unsigned); j>0; j>>=1)

## 6 [P2T]

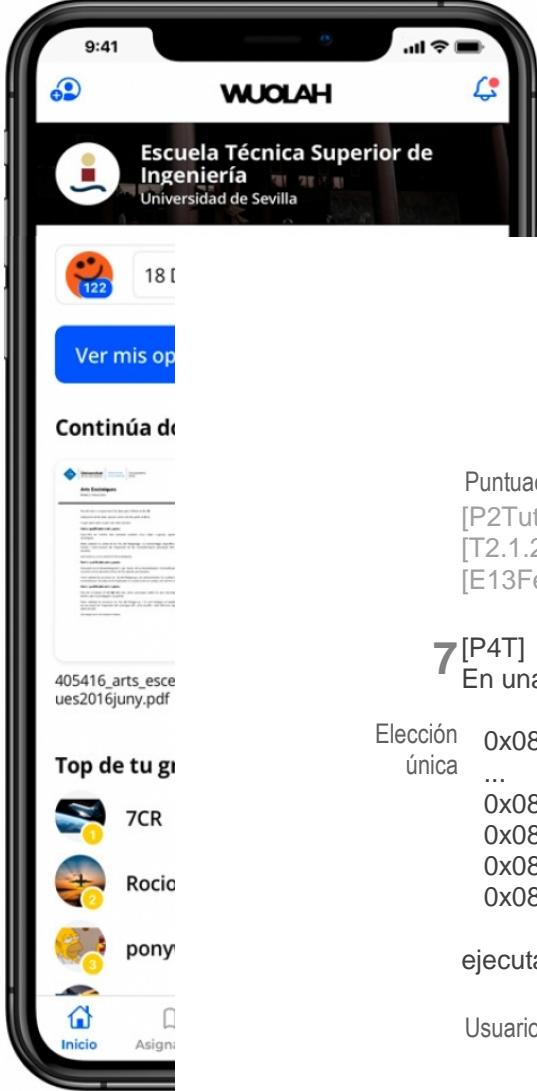
¿Qué hace gcc -O?

Elección  
única

Usuario Profesores



- a) Compilar con optimización suave  
b) Compilar .c→.o (fuente C a objeto)  
c) Compilar .s→.o (fuente ASM a objeto)



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

- d) Ambas (b) y (c), según la extensión de los ficheros que se usen como argumentos

Puntuación: 1,00

[P2Tutorial]  
[T2.1.2Lngjes]  
[E13FebPra02]

7 [P4T]

En una bomba como las estudiadas en prácticas, del tipo...

Elección única 0x08048705 <main+149>: call 0x80484c4 <gettimeofday>  
...

0x08048718 <main+168>: cmp \$0x5,%eax  
0x0804871b <main+171>: jle 0x8048722 <main+178>  
0x0804871d <main+173>: call 0x8048604 <boom>  
0x08048722 <main+178>: ...

ejecutada paso a paso con el depurador ddd, interesaría...

Usuario Profesores

- a) ejecutar hasta jle, ajustar %eax a 6, y continuar ejecutando paso a paso
- b) ejecutar hasta jle, ajustar %eax a 4, y continuar ejecutando paso a paso
- c) cambiar jle por jmp usando ddd o un editor hex,
- d) Ninguna de las opciones anteriores es de interés
- e) salvar el programa y reiniciar la depuración con el nuevo ejecutable
- f) (bien porque no se pueda hacer eso o porque no sirva para evitar la bomba)

Puntuación: 1,00

[P4Tutorial]  
[E13FebPra12]

8 [P5.1]

En la práctica de la cache, el código de line.cc incluye la sentencia

Elección única for (unsigned long long line=1;  
line<=LINE; line<<=1) { ... }

¿Qué objetivo tiene la expresión line<<=1?

Usuario Profesores

- a) volver al principio del vector cuando el índice excede la longitud del vector
- b) duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior

- c) salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
- d) sacar un uno (1) por el stream line

Puntuación: **1,00**

[P5.1Line]

[E15FebPra18]

### **9** [P2.1]

En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando perder acarreos. ¿Cuál es el mínimo valor entero que único repetido en toda la lista causaría acarreo con 32bits (sin signo)?

Usuario Profesores

- a) 0x0800 0000
- b) 0xfbff ffff
- c) 0xfc00 0000
- d) 0x07ff ffff

Puntuación: **1,00**

[P2.1SumUns]

[E14SepPra03]

[E15FebPra14]

[E16FebPra06]

### **10** [P3.2]

La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. La siguiente función contiene un único error (en realidad se han editado 2 líneas de código sobre la versión correcta) pero produce resultado correcto cuando se usa como test el array...

```
int parity4(unsigned* array, int len){
 int i;
 unsigned x;
 int val=0, result=0;

 for (i=0; i<len; i++){
 x = array[i];
 asm("\n"
 "ini: \n\t"
 "xor %[x], %[v] \n\t"
 "shr %[x] \n\t"
 "jnz ini \n\t"
 : [v]"+r" (val)
 : [x] "r" (x)
);
 result += val & 0x1;
 }
}
```

```
 return result;
}
```

Usuario Profesores

- array={1, 16, 256, 1024}  
a) debería salir 4 (todos impares)  
sale 2 (1,0,1,0, cada otro impar deshace el anterior)  
array={1, 2, 4, 8}  
b) debería salir 4 (1,1,1,1, todos impares)  
sale 2 (1,0,1,0, cada otro impar deshace el anterior)  
array={0, 1, 2, 3}  
c) debería salir 2 (0,1,1,0, impares el 1 y el 2)  
sale 1 (0,1,0,0, el 2 deshace el impar del 1)  
array={5, 4, 3, 2}  
d) debería salir 2 (impares el 4 y el 2)  
sale 2 (0,1,1,0, el 3 cuenta como impar, el 2  
deshace el impar)

Puntuación: **0,00**

[P3.2Parity]

[E15SepPra17]

El error consiste en que val=0 se inicializa al principio, en lugar de tras cada x=array[i], de manera que si el LSB de val se queda activado la siguiente paridad se calcula mal. En concreto, tras calcular un impar, todos los pares que sigan cuentan como impar hasta que llegue un impar que deshaga el impar (y entonces el nuevo impar cuenta como par).

## 11 [P4T]

Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la(s) Elección contraseña(s)? Se distingue entre enteros definidos en el código fuente única de la bomba, y enteros solicitados al usuario mediante scanf(). Por "procesar" podemos entender calcular el n-ésimo elemento de la serie de Fibonacci, por ejemplo.

Usuario Profesores

- ✓      •  
a) 1 entero del fuente se procesa, y se compara con el entero del usuario  
b) 2 enteros del usuario se suman, se procesa la suma, y se compara con el entero del fuente  
c) 2 enteros del usuario se procesan, se suman los resultados, y se compara con el entero del fuente  
Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

Puntuación: **1,00**

## 12 [P4T]

En una bomba como las estudiadas en prácticas, del tipo...

Elección única

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>:
```

...el código numérico (pin) es...

Usuario Profesores

- a) el entero almacenado a partir de la posición de memoria 0x24(%esp)
- b) el entero cuya dirección está almacenada en la posición de memoria 0x804a044
- c) el entero 0x804a044
- d) el entero almacenado a partir de la posición de memoria 0x804a044

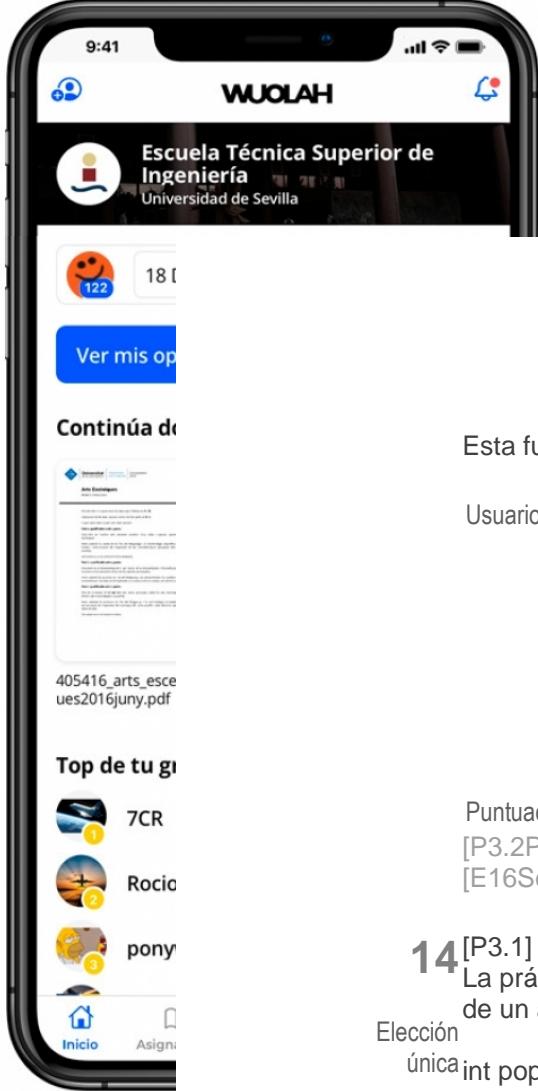
Puntuación: 1,00

## 13 [P3.2]

La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity3:

Elección única

```
int parity3(unsigned* array, int len){
 int i,res=0,val;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
 do {
 val += x;
 x >>= 1;
 } while (x);
 val &= 0x1;
 res+=val;
 }
 return res;
}
```



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## Continúa d

Esta función parity3:

Usuario Profesores

- produce siempre el resultado correcto
- a) solemos escribir res+=val&0x1, en lugar de ponerlo en 2 sentencias C
- b) fallaría con array={0,1,2,3}
- c) fallaría con array={1,2,4,8}
- d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: 0,00

[P3.2Parity]  
[E16SepPra13]

**14**

[P3.1] La práctica "popcount" debía calcular la suma de bits de los elementos de un array. Un estudiante entrega la siguiente versión de popcount4:

```
Elección única
int popcount4(unsigned* array, int len)
{
 int i, j, res = 0;
 for(i = 0; i < len; ++i) {
 unsigned x = array[i];
 int n = 0;
 do {
 n += x & 0x01010101L;
 x >>= 1;
 } while(x);
 for(j = 16; j == 1; j /= 2){
 n ^= (n >>= j);
 }
 res += n & 0xff;
 }
 return res;
}
```

Esta función popcount4:

Usuario Profesores

produce el resultado correcto

Caso real, entregado en prácticas. La máscara está pensada para for(j=0;j<8;j++). En lugar de eso, se hace do...while(x), con lo cual no se ahorran iteraciones y todo el resultado queda acumulado en el LSB de n. El for(j) es absurdo, no itera ninguna vez. El resultado se extrae y acumula con n&0xFF. Es correcto, pero no mejora la eficiencia. popcount2

✓ •

- es igual de eficiente y más elegante, porque no tiene código superfluo.
- b) fallaría con array={0,1,2,3}
  - c) fallaría con array={1,2,4,8}
  - d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: **1,00**

[P3.1Pcount]  
[E13SepPra07]  
[E14FebPra13]  
[E16FebPra09]

En Feb14 se etiquetó incorrectamente como d. Se consideraron válidas a y d.

## 15 [P2T]

Los switches --32 y --64 para trabajar en 32bit/64bit corresponden a la herramienta...

Elección  
única Usuario Profesores

X

- a) ld sería -melf\_i386 y -melf\_xx86-64
- b) as
- c) gcc sería -m32 y -m64
- d) nm no tiene 32/64 bits

Puntuación: **-0,33**

[P2Tutorial]  
[E16SepPra02]

## 16 [P2T]

Dada la siguiente definición de datos:

Elección lista: .int 0x10000000, 0x50000000,  
única 0x10000000, 0x20000000  
longlista: .int (.lista)/4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\0"

La instrucción para copiar la dirección de memoria donde comienza lista en el registro EBX es:

Usuario Profesores

✓

- a) movl \$lista, %ebx  
Práct.2, Tut, págs.8
- b) movl (lista), %ebx
- c) movl \$lista, (%ebx)
- d) movl lista, %ebx

Puntuación: **1,00**

[P2Tutorial]

[E17JulPra05]

## 17 [P2.2]

En la práctica “media” se pide sumar una lista de 32 enteros \*con\* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en único valor absoluto) que repetido en toda la lista causaría overflow con 32bits?

Usuario Profesores

- 0xfbff ffff
- a) en cuanto sea algo más grande que 0xfc00 0000 sale overflow
  - b) 0xfc00 0000
  - c) nx32 == n<<5 y quedaría 0x8000 0000 !!! justo para que no haya overflow
  - d) 0xffff ffff
  - e) eso es -1, y -1x32 == -32 sin problemas
  - f) 0xf000 0000
  - g) se pierde el signo con <<4, mucho más con <<5

Puntuación: **0,00**

[P2.2SumSgn]

[E15SepPra16]

## 18 [P4T]

En una bomba como las estudiadas en prácticas, del tipo...

Elección única

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

Usuario Profesores

- ✓      •      a) el entero 0x804a044
- b) el entero almacenado a partir de la posición de memoria 0x804a044
- c) el string almacenado a partir de la posición de memoria 0x24(%esp)
- d) ninguna de las anteriores

Puntuación: **1,00**

[P4Tutorial]  
[E13FebPra11]  
[E15FebPra17]  
[E16FebPra14]

## 19 [P5.1]

Sea un computador de 32 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:

única

```
int v[262144];
for (i = 0; i < 262144; i += 8)
 v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?

Usuario Profesores

- 1/2 (mitad aciertos, mitad fallos)  
es un array de ints (4B) y se salta  $i+=8$ , los accesos
- a) están separados 32B y las líneas son de 64B, los accesos con índice  $i$  par son fallos, con índice impar son aciertos.
  - b) 1/8 (un fallo por cada 8 accesos)
  - c) 1 (todo son fallos)
  - d) 0 (ningún fallo)

Puntuación: 0,00

[P5.1Line]  
[E17JulPra20]

## 20 [P4T]

En la práctica de la bomba, el primer ejercicio consistía en "saltarse" las "explosiones", para lo cual se puede utilizar...

Elección

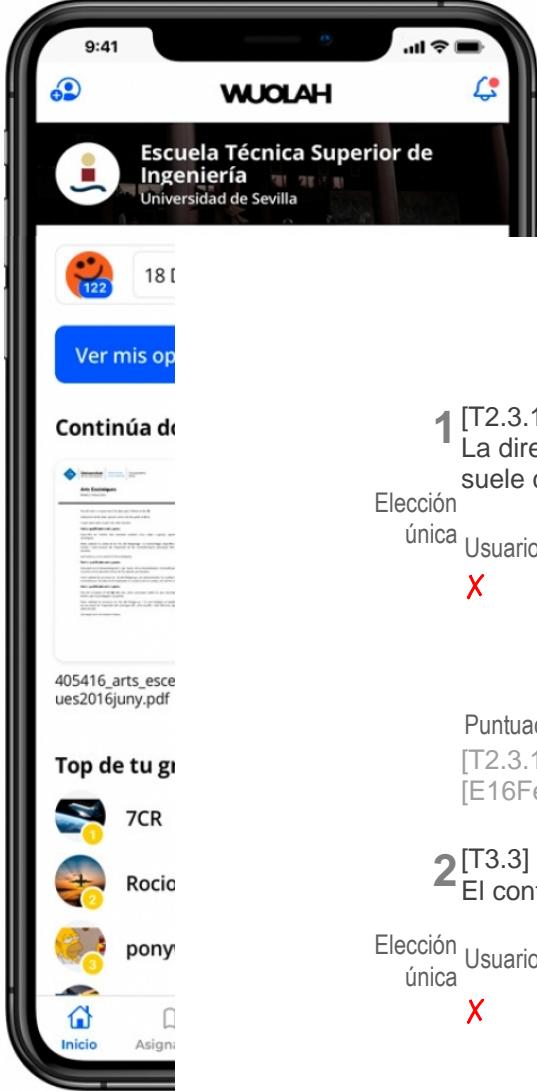
única

Usuario Profesores

- ✓      •      gdb o ddd
- a) un depurador cualquiera sirve para forzar que se cumpla la condición para "saltarse" la bomba
  - b) ddd o hexedit
  - c) hexedit u objdump
  - d) objdump o gdb

Puntuación: 1,00

[P4Tutorial]



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

1 [T2.3.1]

La dirección efectiva del primer parámetro de llamada a una función suele calcularse desde el código de la función como:

Elección  
única  
Usuario Profesores

X

- a) EBP+4
- b) EBP-8
- c) EBP+8
- d) EBP-4

Puntuación: -0,33

[T2.3.1MarcoP]  
[E16FebPra01]

2 [T3.3]

El control residual se utiliza para:

Elección  
única  
Usuario Profesores

X

- a) reducir el tiempo de ejecución de las instrucciones máquina
- b) eliminar los bits residuales de la ejecución de las microinstrucciones
- c) reducir el tamaño de la memoria de control
- d) ninguna de las anteriores es cierta

Puntuación: -0,33

[T3.3CtrlUp]  
[E17JulTeo12]

3 La codificación "Huffman" del código de operación...

Elección  
única  
Usuario Profesores

•

- a) emplea un campo de tamaño fijo para el código de operación de todas las instrucciones.
- b) permite obtener un tamaño promedio del código de operación mínimo.
- c) es la más utilizada.
- d) permite una decodificación muy sencilla de la instrucción.

Puntuación: 0,00

4 [T2.1.4]

En un sistema IA32 Linux, ¿cuál es el tamaño de un long?

Elección  
única  
Usuario Profesores

✓

•

- a) 6 bytes
- b) 4 bytes
- c) 2 bytes
- d) 8 bytes

Puntuación: **1,00**

[T2.1.4x86-64]

[E14FebTeo06]

## 5 [T6.5]

Un computador emplea un sistema de memoria principal de 128 palabras y una memoria cache de 32 palabras. La organización de la memoria cache es totalmente asociativa y el tamaño de bloque es de 8 palabras. Se emplea el algoritmo de reemplazo FIFO. Si inicialmente la memoria cache está totalmente vacía, calcule el número de fallos cuando se lee la secuencia de direcciones de la memoria principal: 0000100, 1000001, 0000101, 0010011, 0100010, 1000100, 0000111.

Elección única Usuario Profesores

- a) 5 fallos
- b) 4 fallos
- c) 3 fallos
- d) 6 fallos

Puntuación: **0,00**

[T6.5MCache]

## 6 [T6.2]

¿En qué tipo de refresco de memoria DRAM CAS# permanece a 0 después del ciclo de lectura o escritura precedente?

Elección

única Usuario Profesores

- a) RAS# antes de CAS#
- b) Sólo RAS#
- c) Refresco transparente
- d) Ninguna de las anteriores respuestas es correcta

Puntuación: **0,00**

[T6.2RAMROM]

## 7 La instrucción movl %esp,%ebp

Elección

única Usuario Profesores

- a) Intercambia los contenidos de los registros ESP y EBP.
- b) Mueve el contenido del registro ESP al registro EBP, poniendo a 0 el registro ESP.
- c) Copia el contenido del registro ESP en el registro EBP.
- d) Introduce en la pila el contenido del registro EBP.

X

Puntuación: **-0,33**

## 8 En las arquitecturas RISC hay...

Elección Usuario Profesores  
única

X

•

- a) pocos registros y muchos tipos de instrucciones.
- b) pocos modos de direccionamiento y muchos formatos de instrucción.
- c) pocas instrucciones muy rápidas con muchos modos de direccionamiento.
- d) muchos registros y pocos modos de direccionamiento.

Puntuación: -0,33

9 Suponiendo que varios dispositivos comparten una única línea de solicitud de interrupción y que varios de ellos solicitan una interrupción al mismo tiempo, ¿qué dispositivo tendría mayor prioridad a la hora de ser atendidas sus peticiones?

única

Usuario Profesores

✓

•

- a) Si se emplea una técnica de sondeo ("polling"), cualquiera de ellos
- Si se emplea una técnica de encadenamiento
- b) ("daisy-chain"), el dispositivo que este más cercano eléctricamente a la CPU
- c) Las respuestas a y b son ciertas
- d) Las respuestas a y b son falsas

Puntuación: 1,00

10 Un Pentium funcionando en modo protegido...

Elección Usuario Profesores  
única

•

- a) siempre tiene activa la unidad de segmentación
- b) siempre tiene activa la unidad de paginación
- c) siempre tiene activas la segmentación y la paginación
- d) puede tener desactivadas la segmentación o la paginación

Puntuación: 0,00

11 ¿Cuál de las siguientes afirmaciones es falsa?

Elección Usuario Profesores  
única

✓

•

- a) La operación de lectura de una celda DRAM es destructiva
- b) Una celda DRAM sólo necesita un transistor y un condensador
- Las memorias DRAM presentan generalmente una
- c) capacidad de almacenamiento mucho mayor que las SRAM
- d) Las memorias DRAM son en general mucho más rápidas que las SRAM

Puntuación: 1,00

## 12 ¿Cuál de las siguientes afirmaciones sobre el direccionamiento absoluto es falsa?

Elección Usuario Profesores  
única

- a) El rango de posiciones direccionables queda limitado por el tamaño del campo de operando.
- b) El objeto está en una posición de la memoria principal.
- c) La instrucción contiene la dirección de memoria exacta en la que se encuentra el objeto.
- d) El tamaño del operando direccionado queda limitado por el nº de bits del campo dirección de memoria.

Puntuación: 1,00

## 13 [T5.3]

¿En qué técnica para determinar la dirección de comienzo de la rutina de servicio de interrupción se fija dicha dirección en los circuitos de la Elección CPU?  
única

Usuario Profesores

- a) Direccionamiento relativo.
- b) Direccionamiento absoluto.
- c) Direcciones fijas.
- d) Envío de instrucción de bifurcación completa.

Puntuación: 1,00

[T5.3ES\_IRQ]

## 14 En el Pentium, el TLB permite buscar rápidamente...

Elección Usuario Profesores  
única

- a) El contenido de un registro caché a partir de un selector
- b) La dirección física a partir de la dirección lineal
- c) La dirección lineal a partir de la dirección virtual
- d) El descriptor de un segmento a partir de un selector

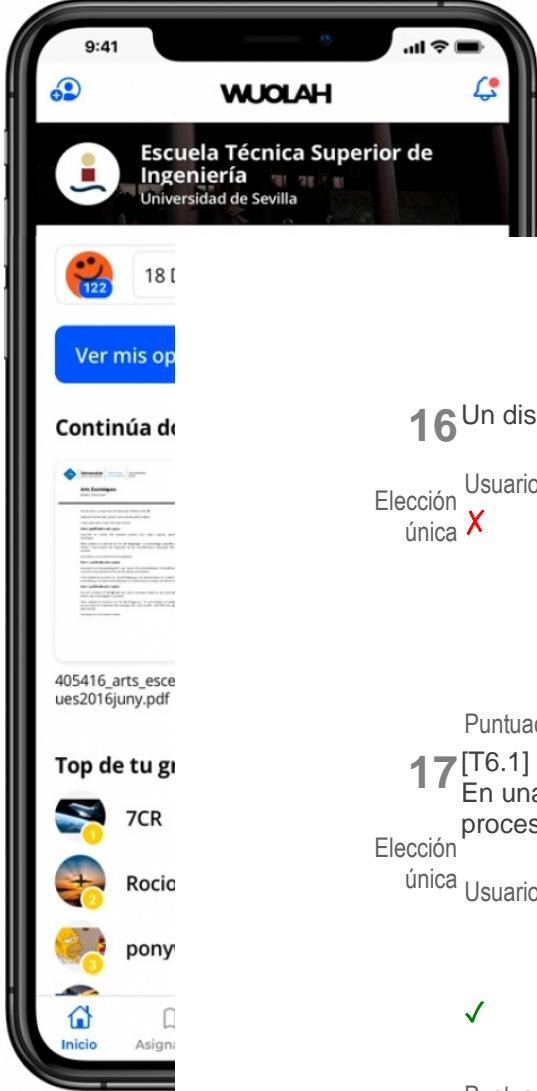
Puntuación: 0,00

## 15 ¿En qué método para determinar la dirección de comienzo de una rutina de servicio de interrupción se envía parte de dicha dirección?

Elección Usuario Profesores  
única

- a) Direccionamiento indirecto
- b) Direccionamiento absoluto
- c) Direccionamiento mediante instrucción de bifurcación
- d) Direccionamiento relativo

Puntuación: 1,00



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

Continúa d

**16** Un diseño vertical de una unidad de control...

Elección única **X** Usuario Profesores

- 
- a) siempre es más rápido que uno horizontal
- b) en general es más lento que uno horizontal
- c) siempre supone un ahorro considerable de bits respecto a uno horizontal
- d) en general desperdicia bits respecto a uno horizontal

Puntuación: **-0,33**

**17** [T6.1]

En una jerarquía de memoria, a medida que nos alejamos del procesador:

Elección única **✓** Usuario Profesores

- 
- 
- a) el tamaño de la memoria disminuye
- b) el tiempo de transferencia disminuye
- c) el tamaño de la unidad de transferencia entre dos niveles aumenta
- d) el coste por byte aumenta

Puntuación: **1,00**

[T6.1ConLoc]

[E12SepTeo29]

**18**

El programador de lenguaje ensamblador necesita conocer:

Elección única **✓** Usuario Profesores

- 
- a) la arquitectura del ordenador.
- b) el diseño RTL del procesador.
- c) todas las respuestas son ciertas.
- d) la microarquitectura del procesador.

Puntuación: **1,00**

**19** [T6.2]

¿Cuál de las siguientes afirmaciones es falsa?

Elección única **✓** Usuario Profesores

- a) La operación de lectura de una celda DRAM es destructiva
- b) Una celda DRAM sólo necesita un transistor y un condensador
- Las memorias DRAM presentan generalmente una
- c) capacidad de almacenamiento mucho mayor que las SRAM
- d) Las memorias DRAM son en general mucho más rápidas que las SRAM

Puntuación: **1,00**

[T6.2RAMROM]

**20** Si queremos almacenar la palabra de 16 bits 9660h en memoria según "little-endian", quedará almacenada a partir de la posición 1000h como:

Elección única Usuario Profesores

- ✓   •   a) en el byte 1000h se guarda 96h y en el 1001h se guarda 60h
- b) en el byte 1000h se guarda 06h y en el 1001h se guarda 69h
- c) en el byte 1000h se guarda 69h y en el 1001h se guarda 06h
- d) en el byte 1000h se guarda 60h y en el 1001h se guarda 96h

Puntuación: **1,00**

**21** A medida que aumenta el tamaño de página en un sistema de memoria virtual, ¿qué ocurre con el tamaño de las tablas de páginas?

Elección única Usuario Profesores

- ✓   •   a) Crece
- b) Disminuye
- c) Permanece constante
- d) Todas las respuestas anteriores son falsas

Puntuación: **-0,33**

**22** [T3.3]

El control residual se utiliza para:

Elección única Usuario Profesores

- X   a) reducir el tiempo de ejecución de las instrucciones máquina
- b) eliminar los bits residuales de la ejecución de las microinstrucciones
- c) reducir el tamaño de la memoria de control
- d) ninguna de las anteriores es cierta

Puntuación: **-0,33**

[T3.3CtrlUp]

**23** [T6.5]

En un sistema con memoria de bytes y líneas de cache de 64 bytes, ¿dónde empieza el bloque de memoria que contiene la posición 0xBEE3DE72?

Elección única

Usuario Profesores

- ✓   •   a) 0xBEE3DE6E
- b) 0xBEE3DE70
- c) 0xBEE3DE40

d) 0x0EE3DE72

Puntuación: **1,00**

[T6.5MCache]

**24** [T2.2.2]

La instrucción xor \$3, %eax tiene como resultado:

Elección única Usuario Profesores



- a) Poner a 0 los últimos 3 bits del registro EAX
- b) Cambiar 0<->1 (complemento a 1 de) los últimos 2 bits del registro EAX
- c) Poner a 1 el último bit del registro EAX
- d) Ninguno de los anteriores resultados

Puntuación: **1,00**

[T2.2.2OpArit]

[E13FebTeo04]

**25** [T2.1.1]

¿Cuál de los siguientes microprocesadores no es de 64 bits?

Elección única Usuario Profesores



- a) Core 2
- b) Itanium
- c) Pentium III
- d) Core i7

Puntuación: **-0,33**

[T2.1.1Histor]

**26**

En una caché asociativa por conjuntos, la vía i está constituida por:

Elección única Usuario Profesores



- a) todos los bloques i-ésimos de cada conjunto
- b) todos los bloques del conjunto i
- c) todos los conjuntos del bloque i
- d) ninguna de las anteriores es cierta

Puntuación: **-0,33**

**27** [T5.3]

La técnica de sondeo, escrutinio o "polling"...

Elección única Usuario Profesores



- a) Se utiliza para identificar el destino de una interrupción
- b) Permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos

- En caso de utilizarse, es necesario emplear varias
- Líneas para que los dispositivos soliciten una interrupción
  - Todas las respuestas anteriores son falsas

Puntuación: **1,00**

[T5.3ES IRQ]

**28** Con una línea de interrupción organizada en colector abierto:

Elección única Usuario Profesores

- a) se pueden conectar varios dispositivos de manera sencilla sin necesidad de circuitos combinacionales
- b) a la patilla del circuito integrado que genera esa línea hay que conectarle un transistor
- c) un dispositivo conectado a ella puede solicitar la interrupción poniendo a nivel bajo la línea
- d) a) y c) son ciertas

Puntuación: **1,00**

**29** [T2.4.1]

Comparando las convenciones de llamada de gcc Linux IA-32 con x86\_64 respecto a registros

Elección única Usuario Profesores

- X
- a) En IA-32 %esi es salva-invocado, y en x86\_64 %rsi es salva-invocado también
  - a) %rsi es 2º argumento - salva-invocante (amarillo en tr.2.4.5)
  - b) En IA-32 %ecx es salva-invocante, y en x86\_64 %rcx es salva-invocante también
  - b) %rcx es 4º argumento - salva-invocante, ok
  - c) En IA-32 %ebx es salva-invocante, pero en x86\_64 %rbx es salva-invocado
  - c) %ebx es salva-invocado (verde en tr.2.3.36)
  - d) En IA-32 %ebp es especial (marco de pila), y en x86\_64 %rbp también
  - d) %rbp es salva-invocado (verde en tr.2.4.5)

Puntuación: **-0,33**

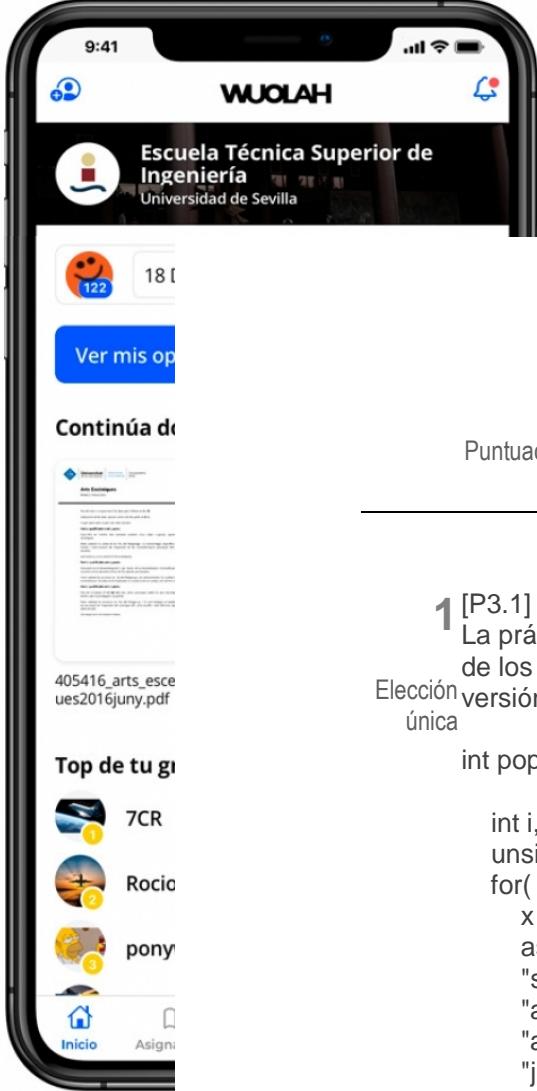
[T2.4.1x86-64]

[E15SepPra11]

**30** ¿En qué orden debería ejecutarse en una máquina de tipo pila la operación aritmética  $(a + b) / (c - d)$ ?

Elección única Usuario Profesores

- $a + b / c - d$
- $a b c d - + /$
- $a b + / c d -$



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

- d) a b + c d - /

Puntuación: 0,00

**1 [P3.1]**  
La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente Elección única versión de popcount3:

```
int popcount3(unsigned* array,
 int len){
 int i, res = 0;
 unsigned x;
 for(i = 0; i < len; i++) {
 x = array[i];
 asm("ini3: \n"
 "shr %[x] \n"
 "adc $0, %[r] \n"
 "add $0, %[x] \n"
 "jne ini3 \n"
 : [r] "+r" (res)
 : [x] "r" (x));
 }
 return res;
}
```

Esta función sólo tiene una diferencia con la versión "oficial" recomendada en clase. En concreto, una instrucción máquina en la sección `asm()` es distinta.

Esta función `popcount3`:

Usuario Profesores

- produce siempre el resultado correcto
- ✓ • a) Si sumar `x+0` activa ZF sólo puede ser porque ya era `x==0`, así que la lógica es equivalente a la deseada.  
b) fallaría con `array={0,1,2,3}`  
c) fallaría con `array={1,2,4,8}`  
d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: 1,00

[P3.1Pcount]  
[E17FebPra11]

Para acabar el bucle cuando `x==0` hubiéramos esperado que se usara

test `%[x]`, `%[x]`, que activa el flag ZF si el resultado es 0. Notar que `and(x,x)` sólo puede ser 0 si ya era `x==0`

## 2 [P5.2]

En la práctica de cache hemos hecho una gráfica con el código `size.cc`  
¿Qué forma tiene la gráfica que se debe obtener?

Elección

única Usuario Profesores

- a) Forma de U (o V) con un tramo descendente y otro ascendente
- b) Una escalera con varios tramos horizontales
- c) Forma de U (o V) invertida, con un tramo ascendente y otro descendente
- d) Forma de /, una gráfica siempre creciente y sin escalones

Puntuación: 1,00

[P5.2Size]

[E14FebPra18]

## 3 [P3T]

En 80x86, los parámetros a las subrutinas se pueden pasar:

Elección

única Usuario Profesores

- a) a través de variables globales
- b) a través de los registros
- c) a través de la pila
- d) todas las anteriores son ciertas

Puntuación: 1,00

[P3Tutorial]

[P2Tutorial]

## 4 [P2T]

¿Qué hace gcc -O?

Elección

única Usuario Profesores

- a) Compilar sin optimización, igual que -O0
- b) Compilar con optimización, igual que -O1
- c) Compilar .c → .o
- d) Compilar .s → .o

Puntuación: 1,00

[P2Tutorial]

[T2.1.2Lngies]

[E13SepPra12]

**5** [P2T]

¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?

Elección

única Usuario Profesores



- a) -m32
- b) -64
- c) -m64
- d) -32

Puntuación: **1,00**

[P2Tutorial]

[E13SepPra14]

**6** [P3T]

¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de una subrutina según la convención cdecl para

Elección

IA32?

única

Usuario Profesores

- a) eax
- b) ebx
- c) ecx
- X d) edx

Puntuación: **-0,33**

[P3Tutorial]

[E14FebPra15]

**7** [P3.1]

¿Cuál es el popcount (peso Hamming, nº de bits activados) del número 19?

Elección

única Usuario Profesores



- a) 2
- b) 3  
19 = 0x13 = 16+3 = 0b0001 0011 -> popcount 3
- c) 4
- d) 5

Puntuación: **1,00**

[P3.1Pcount]

[T2.2.5BWhile]

[E17FebPra10]

**8** [P2T]

Dada la siguiente definición de datos:

lista: .int 0x10000000, 0x50000000,  
0x10000000, 0x20000000

Elección longlista: .int (-lista)/4  
única resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\n0"

la instrucción movl longlista, %ecx copia el siguiente valor:

Usuario Profesores

- a) 32
- b) 4  
Práct.2, Tut, pág.7
- c) 16
- d) 8

Puntuación: **0,00**

[P2Tutorial]  
[E17JulPra06]

**9** [P3T]

¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?

Elección  
única

Usuario Profesores



- Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la variable `ptr=val`
- En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina suponiendo convención cdecl x86, porque x86-64 usa regs.
- Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop no, quita parámetros el invocante, tras retornar de la llamada
- Los parámetros se introducen en la pila en el orden en el que aparecen en la llamada de C, es decir, empezando por el primero y acabando por el último no, de derecha a izquierda

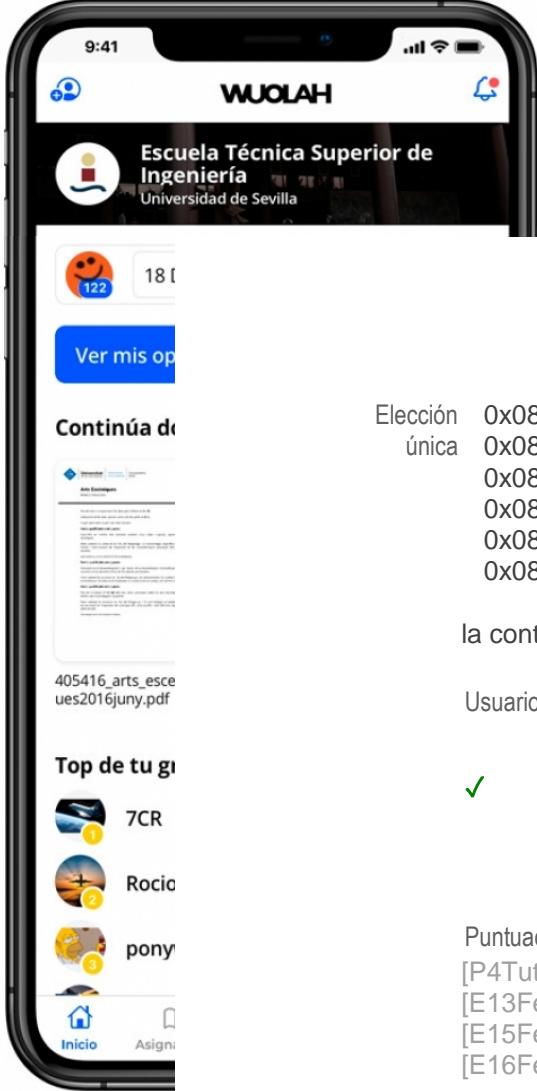
Puntuación: **1,00**

[P2Tutorial]  
[P3Tutorial]  
[E13SepPra16]  
[E17JulPra10]

**10** [P4T]

En una bomba como las estudiadas en prácticas, del tipo...

0x0804873f <main+207>: call 0x8048504 <scanf>



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

Elección única 0x08048744 <main+212>: mov 0x24(%esp),%edx  
 0x08048748 <main+216>: mov 0x804a044,%eax  
 0x0804874d <main+221>: cmp %eax,%edx  
 0x0804874f <main+223>: je 0x8048756 <main+230>  
 0x08048751 <main+225>: call 0x8048604 <boom>  
 0x08048756 <main+230>: ...

la contraseña es...

Usuario Profesores

- a) el entero 0x804a044
- b) el entero almacenado a partir de la posición de memoria 0x804a044
- c) el string almacenado a partir de la posición de memoria 0x24(%esp)
- d) ninguna de las anteriores

Puntuación: 1,00

[P4Tutorial]  
 [E13FebPra11]  
 [E15FebPra17]  
 [E16FebPra14]

11 [P2.1]

Elección única En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32 bits (sin signo)? Se usa notación decimal y espacios como separadores de millares/millones/etc.

Usuario Profesores

- a) 10 000 000
- b) 100 000 000  
No llega, 100 millones < 128M
- c) 1 000 000 000  
Se pasa, 1000 millones >> 128M
- d) 10 000 000 000

Puntuación: -0,33

[P2.1SumUns]  
 [E17FebPra08]

32bit sin signo llega a  $2^{32}-1 = 4G-1 \sim 4\ 000\ 000\ 000$ . Habría acarreo si llegáramos a 4G. Con 32 elementos iguales, tendrían que valer  $4G/32 = 1G/8 = 128M$ .

12 [P3T]

¿En qué registro se pasa el primer argumento a una función en Linux gcc x86-64?

Elección Usuario Profesores

única ✓

- a) edi
- b) edx
- c) esi
- d) ecx

Puntuación: **1,00**

[P3Tutorial]

[T2.4.1x86-64]

[E17JulPra12]

**13** [P4T]

En una bomba como las estudiadas en prácticas, del tipo...

Elección 0x08048705 <main+149>: call 0x80484c4 <gettimeofday>

única ...

0x08048718 <main+168>: cmp \$0x5,%eax

0x0804871b <main+171>: jle 0x8048722 <main+178>

0x0804871d <main+173>: call 0x8048604 <boom>

0x08048722 <main+178>: ...

ejecutada paso a paso con el depurador ddd, interesaría...

Usuario Profesores

- a) ejecutar hasta jle, ajustar %eax a 6, y continuar ejecutando paso a paso
  - b) ejecutar hasta jle, ajustar %eax a 4, y continuar ejecutando paso a paso
  - c) cambiar jle por jmp usando ddd o un editor hex,
  - d) salvar el programa y reiniciar la depuración con el nuevo ejecutable
- Ninguna de las opciones anteriores es de interés
- X d) (bien porque no se pueda hacer eso o porque no sirva para evitar la bomba)

Puntuación: **-0,33**

[P4Tutorial]

[E13FebPra12]

**14** [P3.2]

La práctica “parity” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la

Elección siguiente versión de parity4:  
única

```
int parity4(unsigned* array, int len){
 int val,i,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
```

```

asm("\n"
"ini3: \n\t"
"xor %[x],%[v] \n\t"
"shr %[x] \n\t"
"test %[x], %[x]\n\t"
"jne ini3 \n\t"
:[v]" +r" (val)
:[x] "r" (x)
);
val = val & 0x1;
res+=val;
}
return res;
}

```

La sentencia `asm()` del listado anterior tiene las siguientes restricciones

Usuario Profesores

- a) ninguna
- b) un registro y dos sobreescritos (clobber)
- c) arquitectura de 32 bits
- d) [v] cuenta como salida y entrada, [x] como entrada

Puntuación: **0,00**

[P3.2Parity]

[E16SepPra15]

**15** [P1]

¿De qué tipo son los procesadores Intel que usamos en los laboratorios?

Elección

única

Usuario Profesores



- a) little-endian

el concepto de endian no es aplicable a estas

- b) máquinas, ya que un registro del procesador no cabe en una posición de memoria

- c) big-endian

- d) puede ajustarse mediante un bit de control en el registro CR0 que funcionen como little- o big-endian

Puntuación: **1,00**

[P1]

[T1.1UniFun]

[E15FebPra01]

**16** [P1]

La etiqueta del punto de entrada a un programa ensamblador en el entorno de las prácticas 1 a 4 (GNU/as Linux x86) es:

Elección Usuario Profesores

única **X**

- a) \_\_init
- b) \_\_main
- \_\_start
- P3 y P4 se redactan en C.
- P1 y P2 sí son en ensamblador GNU/as Linux x86.
- c) Incluso en P2 se llega a usar main para ensamblar con gcc ya que usamos printf.  
En cualquier caso, las otras opciones son descabelladas
- d) .LO

Puntuación: **-0,33**

[P1]

[E17FebPra03]

**17** [T2.1.2]

En X86-64, el registro contador de programa se denomina:

Elección Usuario Profesores

única **✓**

- a) RIP
- b) R15
- c) EIP
- d) IP

Puntuación: **1,00**

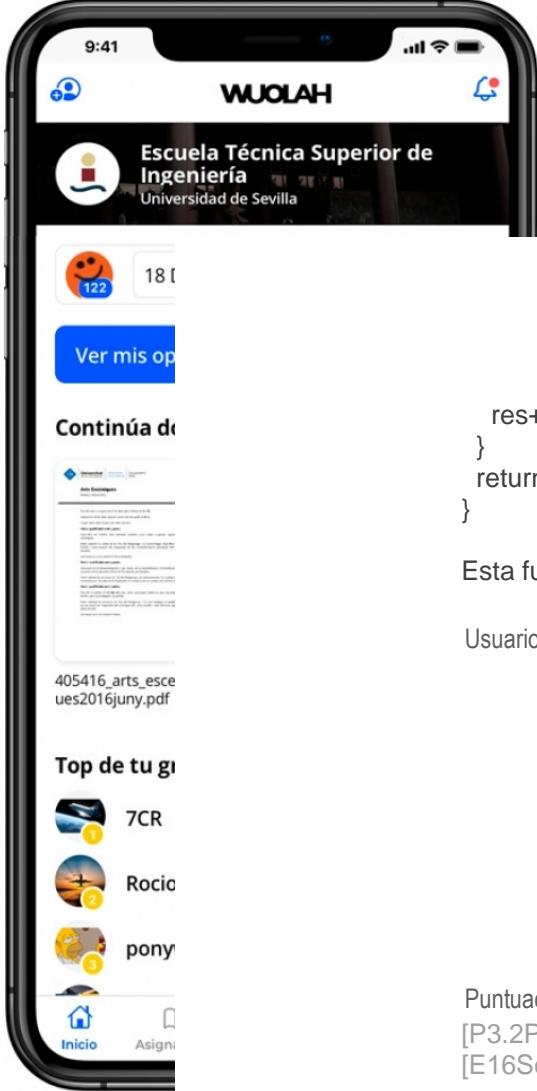
[T2.1.2Lngjes]

[E16FebTeo03]

**18** [P3.2]

La práctica “parity” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la Elección siguiente versión de parity4:  
única

```
int parity4(unsigned* array, int len){
 int val,i,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
 asm("\n"
 "ini3: \n\t"
 "xor %[x],%[v] \n\t"
 "shr %[x] \n\t"
 "test %[x], %[x]\n\t"
 "jne ini3 \n\t"
 "[v]"+"r" (val)
 "[x]" "r" (x)
);
 val = val & 0x1;
```



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

```
res+=val;
}
return res;
}
```

Esta función parity4:

Usuario Profesores

produce siempre el resultado correcto  
solemos escribir res+=val&0x1, en lugar de ponerlo

- a) en 2 sentencias C  
adicionalmente, shr afecta al flag ZF de manera que sobra test %[x],%[x]
- b) fallaría con array={0,1,2,3}
- c) fallaría con array={1,2,4,8}
- d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: 0,00

[P3.2Parity]  
[E16SepPra14]

**19** [P2T]

Dada la siguiente definición de datos:

Elección lista: .int 0x10000000, 0x50000000,  
única 0x10000000, 0x20000000  
longlista: .int (. -lista)/4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\0"

la llamada correcta a printf será:

Usuario Profesores

```
push resultado
push resultado+4
push resultado
push resultado+4
```

- a) push \$formato  
call printf  
add \$20, %esp  
No. Debe ser little-endian, los push resultado están desordenados
- ```
push resultado+4
push resultado
```
- b) push \$formato
call printf
add \$12, %esp
No. 2 veces resultado

WUOLAH

- ✓ •
- c) push resultado
push resultado+4
push \$formato
call printf
add \$12, %esp
No. 2 veces resultado
push resultado+4
push resultado
push resultado+4
 - d) push resultado
push \$formato
call printf
add \$20, %esp
- Sí. $20 = 8 (\%llx) + 8 (\%llu) + 4 (\$formato)$

Puntuación: **1,00**

[P2Tutorial]

[E17JulPra08]

notar que formato lleva %llu y %llx, para imprimir (se adivina que) dos veces resultado

20 [P2T]

¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?

Elección

única Usuario Profesores

- ✓ •
- a) -march64
 - b) -m32
 - c) -march32
 - d) -m64

Puntuación: **1,00**

[P2Tutorial]

[E17FebPra02]

1

¿En qué técnica para determinar la dirección de comienzo de la rutina de servicio de interrupción se fija dicha dirección en los circuitos de la CPU?

Elección

única Usuario Profesores

- ✓ •
- a) Direccionamiento relativo.
 - b) Direccionamiento absoluto.
 - c) Envío de instrucción de bifurcación completa.
 - d) Direcciones fijas.

Puntuación: **1,00**

2 [T5.2]

La E/S programada:

Elección Usuario Profesores
única



- Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque una instrucción de transferencia individual de datos con la interfaz del periférico (por ej. IN, OUT) es más lenta en E/S programada que en E/S por interrupciones.
- Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU es más rápida que el controlador de interrupciones y la interfaz del periférico.
- Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU debe encargarse de la sincronización con la interfaz del periférico haciendo una espera activa.
- Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU tiene el control de toda la operación.

Puntuación: **1,00**

[T5.2ESProg]

[T5.3ES_IRQ]

[E17FebTeo21]

3 ¿Cuál de las siguientes afirmaciones es falsa?

Elección Usuario Profesores
única



- a) la anchura del bus de datos es siempre de 16 bytes
- b) el bus de datos es bidireccional
- c) el bus de direcciones es unidireccional
- d) el bus de control puede transportar señales de estado

Puntuación: **1,00**

4 [T5.1]

Algunas de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente son:

Elección Usuario Profesores
única



- a) Los puertos de E/S no ocupan direcciones de memoria.
- b) Facilita la protección de E/S.
- c) El diseño de la CPU es más sencillo.
- d) Todas las respuestas anteriores son ciertas.

Puntuación: **-0,33**

[T5.1FunE/S]

5 La "postescritura (write-back) marcada"

- Elección única Usuario Profesores
- a) es más eficiente que la "postescritura siempre"
b) requiere más bits de modificación ("bits sucios")
c) cuando aumenta el número de vías
d) provoca una menor tasa de faltas que la "postescritura siempre"

Puntuación: 0,00

6 [T6.3]

Con 8 circuitos de memoria RAM de 1K x 8 se puede crear un memoria de:

- Elección única Usuario Profesores
- a) 1K x 64
b) 8K x 8
c) 2K x 32
✓ • d) Todas las combinaciones anteriores son posibles

Puntuación: 1,00

[T6.3Diseño]

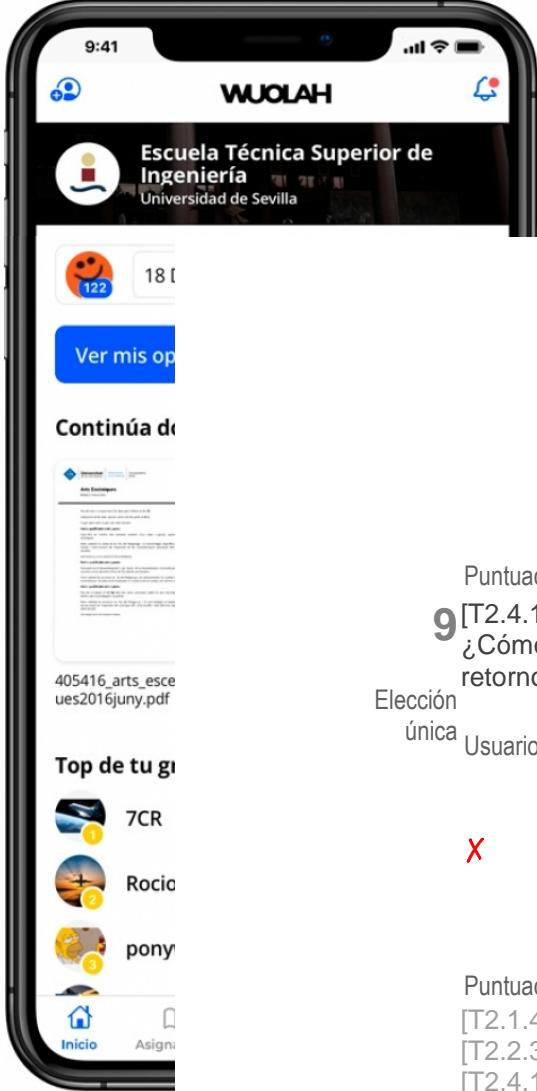
7 La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...

- Elección única Usuario Profesores
- ✓ • a) dos buffers triestado
b) tres conexiones directas al bus común
c) dos multiplexores de 2 a 1
d) tres demultiplexores

Puntuación: 1,00

8 Respecto a las unidades de control nanoprogramadas:

- Elección única Usuario Profesores
- a) La realización nanoprogramada de una unidad de control es más rápida que la micropogramada.
La anchura de la memoria de nanoprograma es la misma que la de memoria de micropograma en un diseño de la misma unidad de control que no usara nanoprogramación.
c) El diseño de las unidades de control nanoprogramadas debe ser vertical.



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

Suponiendo una memoria de microprograma con n microinstrucciones de w bits cada una, de las cuales 2^m son distintas, el ahorro en bits si se utiliza nanoprogramación es $(n \cdot m + 2^m \cdot w) - n \cdot w$.

Puntuación: 0,00

9 [T2.4.1]

¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una función long int al terminar ésta?

Elección única Usuario Profesores

X

- a) La instrucción RET lo almacena en un registro especial de retorno.
- b) Por convención se guarda en %eax.
- c) Se almacena en pila justo encima de los argumentos de la función.
- d) Ninguna de esas formas es la correcta.

Puntuación: -0,33

[T2.1.4x86-64]
[T2.2.3CodCon]
[T2.4.1x86-64]
[P3Tutorial]
[E12SepTeo02]

10 [T2.3.1]

¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?

Elección única Usuario Profesores

✓

•

- Antes de volver de la rutina llamada, el programa
- a) en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop
 - b) Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero
 - c) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable
 - d) En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

Puntuación: 1,00

[T2.3.1MarcoP]
[P3Tutorial]
[E13FebPra06]

11 [T6.2]

¿En qué tipo de ciclo de refresco se hace RAS# = 0?

Elección única Usuario Profesores

- a) Sólo RAS#
- b) CAS# antes de RAS#
- c) Refresco transparente
- d) En todos los anteriores

Puntuación: **0,00**

[T6.2RAMROM]

12 [T6.3]

Una puerta AND con 16 entradas conectada a un bus de direcciones de 16 bits, con todos los bits negados excepto A10 y A6, permite seleccionar un dispositivo (con CS activa en alta) en la dirección:

Usuario Profesores

X

- a) 0x0220
- b) 0xFBFF
- c) 0x0440
- d) 0xFDDF

Puntuación: **-0,33**

[T1.3EstBus]

[T6.3Diseño]

[E17FebTeo22]

13

Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

Elección única Usuario Profesores

- a) esi
- b) ebx
- c) edi
- d) eax

Puntuación: **1,00**

14

Señale cuál de las siguientes opciones no es un modo para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

Elección única Usuario Profesores

- a) E/S programada
- b) E/S por flanco
- c) Acceso directo a memoria (DMA)
- d) E/S iniciada por interrupción

Puntuación: **1,00**

15 [T3.3]

Un procesador con una unidad de control microprogramada tiene una memoria de control de 300 palabras de 100 bits, de las que 200 son diferentes. Si se rediseñara como unidad de control nanoprogramada, ¿qué tamaño ocuparía la nanomemoria que contiene las microinstrucciones completas sin repeticiones?

Usuario Profesores

- ✓ • a) 22400 bits
b) 21600 bits
c) 30000 bits
d) 20000 bits
✓ • d) 200 uinstr. x 100 bits

Puntuación: **1,00**

[T3.3CtrlUp]

[E17FebTeo15]

16

Un controlador de E/S posee un buffer para el almacenamiento temporal de los datos con una capacidad de 64 KB. En un instante determinado inicia una operación de E/S con una impresora a una velocidad de transferencia de 128 KB/s. Si el controlador de E/S recibe la información que debe enviar a la impresora a una velocidad de 1 MB/s, ¿cuánto tiempo tardará en llenarse por primera vez el buffer suponiendo que inicialmente está vacío, y que recibe y envía información simultáneamente de forma continua?

Usuario Profesores

- ✓ • a) 71 ms
b) 143 ms
c) No se puede calcular
d) Ninguna de las anteriores

Puntuación: **1,00**

17

¿Es posible utilizar 4 GB de memoria en un sistema cuya CPU emplea E/S mapeada en memoria, cuyo bus de direcciones es de 32 bits y que tiene al menos un puerto de E/S? Supondremos que no se puede emplear ninguna técnica de extensión del bus de direcciones.

Usuario Profesores

- X • a) Sí
b) No
c) Depende de si el número puertos de E/S es muy elevado
d) Ninguna de las respuestas anteriores es correcta

Puntuación: **-0,33**

18 [T2.2.1]

¿Cuál de las siguientes instrucciones máquina copia en EAX el entero

Elección almacenado en la posición de memoria cuya dirección efectiva es el único resultado de la operación EDX*4 + EBX?

Elección única Usuario Profesores

- X a) movl 4(%edx, %edx), %eax
• b) leal (%ebx, %edx, 4), %eax
c) movl (%ebx, %edx, 4), %eax
d) leal 4(%edx, %edx), %eax

Puntuación: -0,33

[T2.2.1ModDir]

[P2Tutorial]

[E17JulPra02]

19 ¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?

Elección única Usuario Profesores

- ✓ • a) Registro contador
b) Registro de prioridades
c) Registro de órdenes
d) Registro de dirección

Puntuación: 1,00

20 [T2.2.3]

La instrucción IA32 test sirve para...

Elección única Usuario Profesores

- ✓ • a) Realizar la operación resta (a-b) pero no guardar el resultado, sino simplemente ajustar los flags
b) Mover el operando fuente al destino, pero sólo si se cumple la condición indicada
Realizar la operación and lógico bit-a-bit (a&b) pero
c) no guardar el resultado, sino simplemente ajustar los flags
d) Testear el código de condición indicado, y poner un byte a 1 si se cumple

Puntuación: 1,00

[T2.2.3CodCon]

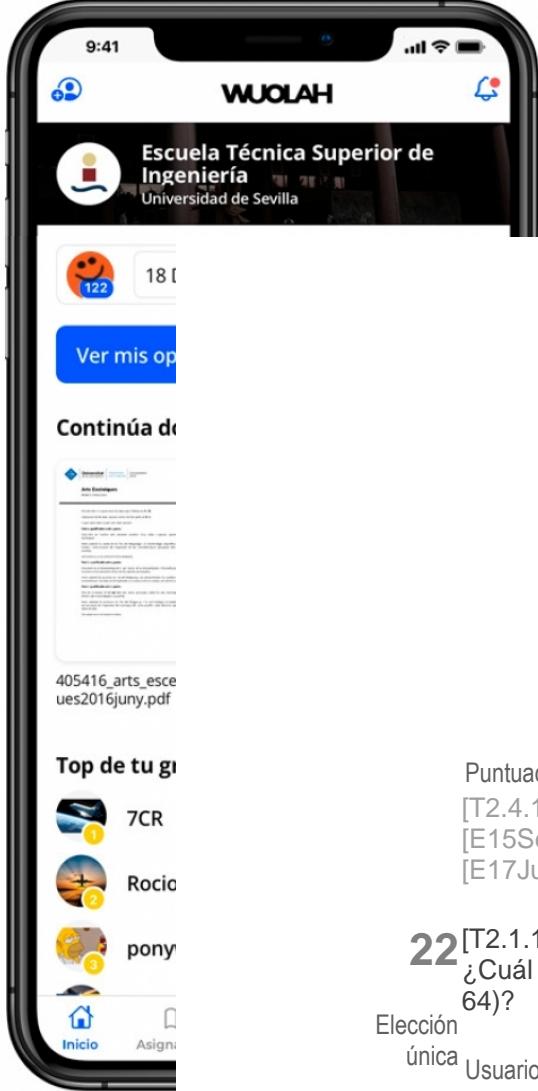
[E13SepTeo04]

21 [T2.4.1]

Comparando las convenciones de llamada de gcc Linux IA32 con x86-64 respecto a registros

Elección única Usuario Profesores

- X a) En IA32 %esi es salva-invocado, y en x86-64 %rsi es salva-invocado también



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

%rsi es 2º argumento - salva-invocante (amarillo en tr.2.4.5)

En IA32 %ecx es salva-invocante, y en x86-64

- b) %rcx es salva-invocante también

%rcx es 4º argumento - salva-invocante, ok

En IA32 %ebp es especial (marco de pila), y en

- c) x86-64 %rbp también

%rbp es salva-invocado (verde en tr.2.4.5)

En IA32 %ebx es salva-invocante, pero en x86-64

- d) %rbx es salva-invocado

%ebx es salva-invocado (verde en tr.2.3.36)

Puntuación: **-0,33**

[T2.4.1x86-64]

[E15SepPra11]

[E17JulTeo06]

22 [T2.1.1]

¿Cuál fue el primer procesador de Intel de 64-bits en la familia x86(-64)?

Elección única
Usuario Profesores

- ✓ • a) Pentium 4F
b) 386
c) Core i7
d) 8086

Puntuación: **1,00**

[T2.1.1Histor]

[E14SepTeo01]

23 [T6.1]

Sea un computador de 32 bits que dispone de una memoria cache de 512 KB y líneas de 64 bytes. ¿Cuántas líneas tiene la cache?

Elección única
Usuario Profesores

- a) 8192
b) 1024
c) 65536
X d) 64

Puntuación: **-0,33**

[T6.1ConLoc]

[E14FebTeo20]

24 [T2.3.1]

En la secuencia de programa siguiente:

804854e:e8 3d 06 00 00 call 8048b90 <main>

8048553:50 pushl %eax

Elección

única ¿cuál es el valor que introduce en la pila la instrucción call?

Usuario Profesores

- a) 0x8048b90
- b) 0x804854f
- c) 0x8048553
- d) 0x804854e

Puntuación: **0,00**

[T2.3.1MarcoP]

[E16FebTeo14]

25 [T5.2]

¿Cuántos puertos de E/S permite manejar la interfaz de periféricos programable 8255?

Elección

única Usuario Profesores

- a) 2 puertos de 16 bits
- b) 3 puertos de 8 bits
- c) 4 puertos de 32 bits
- d) Todas las respuestas anteriores son falsas

Puntuación: **0,00**

[T5.2ESProg]

26

Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para escribir el resultado en el banco de registros, ¿cuál es la frecuencia de reloj máxima del procesador?

Usuario Profesores

- a) 200 MHz
- b) 66,67 MHz
- c) 500 MHz
- d) 40 MHz

Puntuación: **0,00**

27 [T5.1]

Respecto a si un computador dispone de E/S independiente (separada) o usa E/S mapeada a memoria:

Elección

única Usuario Profesores

- Si el encapsulado del procesador no dispone de
- a) patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada
 - ✓ • b) Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada

- Si el repertorio del procesador tiene instrucciones
c) del tipo LOAD y STORE, el computador sólo
dispone de E/S mapeada a memoria
Si el encapsulado (chip) del procesador tiene patilla
d) (pin) IO/M# (o patillas equivalentes), eso evidencia
que el computador usa E/S mapeada a memoria

Puntuación: **1,00**

[T5.1FunE/S]

[E15FebTeo16]

28 [T2.2.4]

Las instrucciones JB y JNAE del Pentium provocan un salto si...

Elección única Usuario Profesores

- a) ZF == 0
- b) CF == 1
- c) ZF != SF
- d) SF == 1

X Puntuación: **-0,33**

[T2.2.4SalCon]

29 [T5.2]

¿Cuántos puertos puede gestionar la interfaz de periféricos
programable 8255?

Elección única Usuario Profesores

- a) Uno de 24 bits
- b) Dos de 12 bits
- c) Tres de 8 bits
- d) Todas las combinaciones anteriores son válidas

Puntuación: **0,00**

[T5.2ESProg]

30

Utilizando E/S programada y como modo de direccionamiento
selección lineal, ¿cuántos periféricos podrían conectarse a un 8086?

Elección única Usuario Profesores

- a) 20 periféricos
- b) 16 periféricos
- c) Cualquier número de periféricos
- d) 8 periféricos

Puntuación: **0,00**

1 [P3T]

¿En qué registro se pasa el primer argumento a una función según el estándar _cdecl en una arquitectura IA32?

Elección

única

Usuario Profesores

X

- a) edi
- b) esi
- c) eax
- d) Las anteriores respuestas son erróneas

Puntuación: **-0,33**

[P3Tutorial]

[T2.4.1x86-64]

[E15FebPra02]

2 [P2.2]

En la práctica “media” se pide sumar una lista de enteros *con* signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en los únicos dos primeros elementos de la lista causaría overflow con 32 bits al realizar la suma de esos dos primeros elementos de la lista?

Usuario Profesores

- a) 0x0400 0000
suma 0x0800 0000 sin problema
- b) 0x8000 0000
es negativo, el enunciado pide menor valor positivo
- c) 0x0800 0000
suma 0x1000 0000 sin problema
- ✓ • d) 0x4000 0000
0x8000 0000 sería negativo en 32bit

Puntuación: **1,00**

[P2.2SumSgn]

[E17JulPra09]

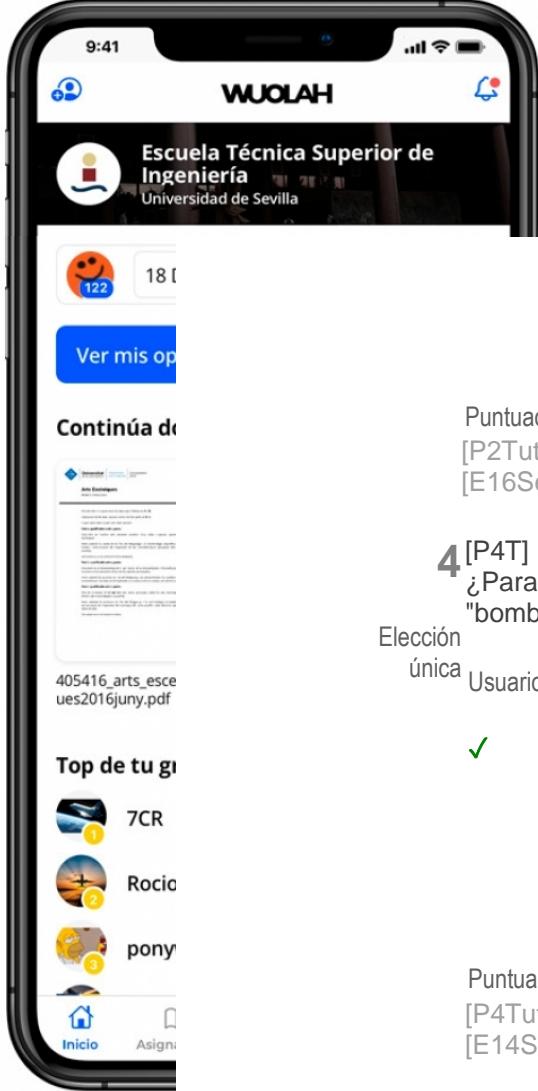
3 [P2T]

El switch de gcc para que únicamente compile de lenguaje C a ensamblador, y no realice ningún paso adicional (ensamblar, enlazar, etc), es...

única

Usuario Profesores

- a) -g
para incorporar info depuración
- b) -c
de C/asm a objeto, .c/.s→.o
- c) -o
para nombrar el ejecutable
- ✓ • d) -S



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Puntuación: 1,00

[P2Tutorial]
[E16SepPra01]

4 [P4T]

¿Para qué se utiliza la función gettimeofday en la práctica de la "bomba digital"?

Elección
única Usuario Profesores

- Para lanzar un error cuando el usuario tarde
- a) demasiado tiempo en introducir la contraseña o el PIN
 - b) Para cronometrar y poder comparar las duraciones de las distintas soluciones del programa
 - c) Para cifrar la contraseña en función de la hora actual
 - d) Para imprimir la hora en la pantalla

Puntuación: 1,00

[P4Tutorial]
[E14SepPra20]

5 [P2T]

La siguiente línea en la sección de datos de un programa en ensamblador de IA32

Elección
única result: .int 0,0

Usuario Profesores

- a) Reserva espacio para un único entero, inicializado a 0,0
- b) Reserva espacio para un único entero, inicializado a 0, en la posición de memoria 0
- c) Reserva espacio para un entero, inicializado a 0,
- d) Reserva espacio para dos enteros, inicializados ambos a 0

Puntuación: 1,00

[P2Tutorial]
[E17FebPra04]

6 [P2A2]

¿Cuál de los siguientes grupos de instrucciones IA32 sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

Elección
única Usuario Profesores

- a) IMUL, IDIV
- b) AND, OR, XOR

- ✓ • c) CMP, TEST
d) ADC, SBB

Puntuación: **1,00**

[P2Apéndice2]

[T2.2.2OpArit]

[T2.2.3CodCon]

[E12FebTeo16]

7 [P4T]

En la práctica de la bomba necesitamos estudiar el código máquina de la bomba del compañero. A veces dicho código no se visualiza directamente en el depurador ddd, y algunas de las técnicas que se única pueden probar para conseguir visualizarlo son... (marcar la opción *falsa*)

Usuario Profesores

- X
- a) comprobar que está activado el panel View → Machine Code Window
 - b) asegurarse de que se ha escrito correctamente el nombre del ejecutable
 - c) escribir info line main en el panel de línea de comandos gdb
recompilar con información de depuración, por si se nos había olvidado, ya que sin -g el ejecutable no
 - d) contiene información de depuración
sí, claro, como que nos van a pasar el código fuente... :-)

Puntuación: **-0,33**

[P4Tutorial]

[E17JulPra14]

8 [P1]

El punto de entrada de un programa ensamblador en GNU/as Linux x86 se llama

Elección

única Usuario Profesores

- ✓ • a) _init
b) main
c) _start
d) begin

Puntuación: **1,00**

[P1]

[E14FebPra06]

9 [P2.2]

En la práctica “media” se pide sumar una lista de 32 enteros *con* signo de 32bits en una plataforma de 32bits sin perder precisión, esto

Elección es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en única valor absoluto) que repetido en toda la lista causaría overflow con 32bits?

Usuario Profesores

- a) 0xf000 0000
se pierde el signo con <<4, mucho más con <<5
- b) 0xffff ffff
eso es -1, y $-1 \times 32 == -32$ sin problemas
0xfbff ffff
- c) en cuanto sea algo más grande que 0xfc00 0000 sale overflow
0xfc00 0000
- d) $n \times 32 == n \ll 5$ y quedaría 0x8000 0000 !!! justo para que no haya overflow

Puntuación: 1,00

[P2.2SumSgn]

[E15SepPra16]

10 [P3T]

En la convención cdecl estándar para arquitecturas x86 de 32 bits, cuál de las siguientes afirmaciones es cierta:

Elección

única

Usuario Profesores

- a) Los parámetros se pasan en pila, de derecha a izquierda; es decir, primero se pasa el último parámetro, después el penúltimo... y por fin el primero
- b) Solamente es necesario salvar el registro EAX
- c) Los registros EBX, ESI y EDI son salva-invocante
- d) Ninguna de las anteriores es cierta

Puntuación: 1,00

[P3Tutorial]

[E14FebPra01]

11 [P2.2]

En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32bits (en complemento a 2)?

Elección

única

Usuario Profesores

- a) 0xf800 0001
- b) 0xfc00 0000
- c) 0xf800 0000
- d) 0xfbff ffff

Puntuación: **1,00**

[P2.2SumSgn]
[E14SepPra04]
[E16FebPra07]

12 [P2T]

Los switches --32 y --64 para trabajar en 32bit/64bit corresponden a la herramienta...

Elección
única Usuario Profesores

- a) nm
no tiene 32/64 bits
- b) gcc
sería -m32 y -m64
- c) ld
sería -melf_i386 y -melf_xx86-64
- ✓ d) as

Puntuación: **1,00**

[P2Tutorial]
[E16SepPra02]

13 [P4T]

Una de las “bombas” utiliza el siguiente código para cifrar la clave numérica introducida por el usuario y ahora almacenada en eax:

Elección
única
804870d: xor \$0xffff,%eax
8048712: mov \$0x2,%ecx
8048717: cltd
8048718: idiv %ecx
804871a: cmp %eax,0x804a034

Si el entero almacenado a partir de 0x804a034 es 0x7ff, la clave numérica puede ser:

Usuario Profesores

- a) 0x1009 4F97 (269 045 655)
- b) 0xffff (4095)
- X c) 0x7ff (2047)

1

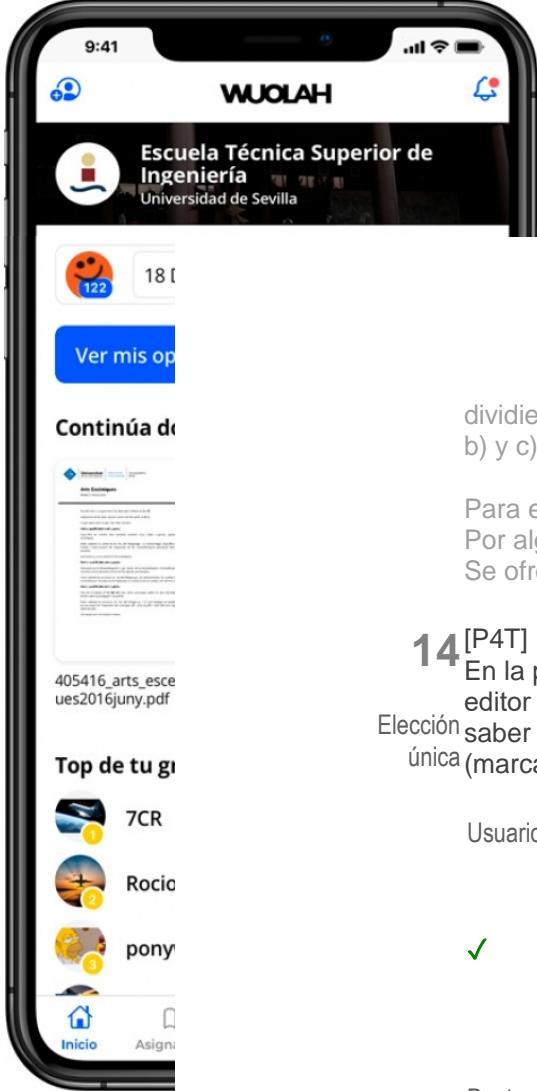
Tanto 0 como 1 servirían de contraseña.

- d) El enunciado de la bomba prohibía explícitamente casos como éste.
La bomba debe ser un valor fijo. Uno, no dos.

Puntuación: **-0,33**

[P4Tutorial]
[E17FebPra18]

El código invierte los últimos 12 bits de la clave numérica introducida por el usuario, pierde entonces el LSB (bit menos significativo)



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

dividiendo por 2, y lo que quede debe coincidir con 0x7ff (las opciones b) y c) quedan automáticamente descartadas).

Para ello el número original debería haber sido 0 ó 1.

Por algún motivo, es fácil confundirse razonando.

Se ofrecen opciones que eviten confundirse con 0xffff f000 y similares.

14 [P4T]

En la práctica de la bomba, el tercer ejercicio consistía en usar un editor hexadecimal para crear un ejecutable sin "explosiones". Para saber qué contenidos del fichero hay que modificar, se puede utilizar... única (marcar la opción *falsa*)

Usuario Profesores

- a) objdump
hexedit
- ✓ b) el enunciado implícitamente indica que no basta con hexedit. Aún así hubo quien la dejó en blanco (pocos) y quien falló (50%)
- c) gdb
- d) ddd

Puntuación: 1,00

[P4Tutorial]

[E17JulPra17]

15 [P3.2]

La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned * array, int len)
{
    int i, result = 0;
    unsigned x;
    for (i=0; i<len; i++){
        x = array[i];
        asm( "mov %[x], %%edx \n\t"
             "shr $16, %%edx \n\t"
             "shr $8, %%edx \n\t"
             "xor %%edx,%%edx \n\t"
             "setp %%dl    \n\t"
             "movzx %%dl,%[x] \n\t"
             : [x] "+r" (x)
             :
             : "edx"
            );
        result += x;
    }
    return result;
}
```

}

Esta función parity6:

Usuario Profesores

- a) produce el resultado correcto
no es correcta; fallaría por ejemplo con
array={0,1,2,3}
Caso real, entregado en prácticas. Las tres primeras instrucciones asm se pierden al poner edx
- b) a 0 usando xor. Consecuentemente, se activa PF para ajustar a impar, y termina siendo x=1. Es decir, todos los elementos del array contabilizan paridad=1. El array {1,2,4,8} pasa desapercibido, pero {0,1,2,3} debería producir resultado=2<>4.
- c) no es correcta; fallaría por ejemplo con
array={1,2,4,8}
- d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Puntuación: **0,00**

[P3.2Parity]
[E13SepPra08]
[E14FebPra14]
[E15FebPra15]
[E16FebPra10]

16 [P2T]

Como parte del proceso de compilación de una aplicación en lenguaje C, enlazar .o → .exe (de objeto proveniente de fuente C a ejecutable)
Elección usando sólo as y ld, sin gcc...
única

Usuario Profesores

- a) Ninguna de las anteriores respuestas es correcta
- b) Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
- c) Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C
- X d) Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C

Puntuación: **-0,33**

[P2Tutorial]
[E13SepPra06]

17 [P3T]

Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%r"
    :[r] "+r" (result)
    :[i] "r" (i),
```

Elección [a] "r" (array));
única ¿Cuál de las siguientes afirmaciones es correcta?

Usuario Profesores

- a) la salida de la función se fuerza a que esté en la variable result
- b) r es una posición de memoria de entrada/salida
- c) i es un registro de entrada
- d) a es una posición de memoria de entrada

X

Puntuación: -0,33

[P3Tutorial]
[E13FebPra10]

18 [P5.1]

En la práctica de la cache, el código de line.cc incluye la sentencia

Elección for (unsigned long long line=1;
única line<=LINE; line<<=1) { ... }

¿Qué objetivo tiene la expresión line<<=1?

Usuario Profesores

- a) sacar un uno (1) por el stream line
- b) salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
- c) volver al principio del vector cuando el índice exceda la longitud del vector
- d) duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior

✓ • Puntuación: 1,00

[P5.1Line]
[E14FebPra19]

19 [P3T]

¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?

Elección Usuario Profesores
única

- Pasar a una función un puntero a una variable se
- a) traduce en introducir en la pila el valor de la variable
 - b) En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
 - c) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop

✓ •

- Los parámetros se introducen en la pila en el orden
d) en el que aparecen en la llamada de C, es decir,
empezando por el primero y acabando por el último

Puntuación: **1,00**

[P2Tutorial]

[P3Tutorial]

[E13SepPra16]

20 [P2.1]

En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo único valor entero que repetido en toda la lista causaría acarreo con 32 bits (sin signo)? Se usa notación decimal y espacios como separadores de millares/millones/etc.

Usuario Profesores

- a) 10 000 000
- b) 100 000 000
No llega, 100 millones < 128M
- ✓ c) 1 000 000 000
Se pasa, 1000 millones >> 128M
- d) 10 000 000 000

Puntuación: **1,00**

[P2.1SumUns]

[E17FebPra08]

32bit sin signo llega a $2^{32}-1 = 4G-1 \approx 4\ 000\ 000\ 000$.

Habría acarreo si llegáramos a 4G. Con 32 elementos iguales, tendrían que valer $4G/32 = 1G/8 = 128M$.

Puntuación: **12,33 (6,17 sobre 10)**

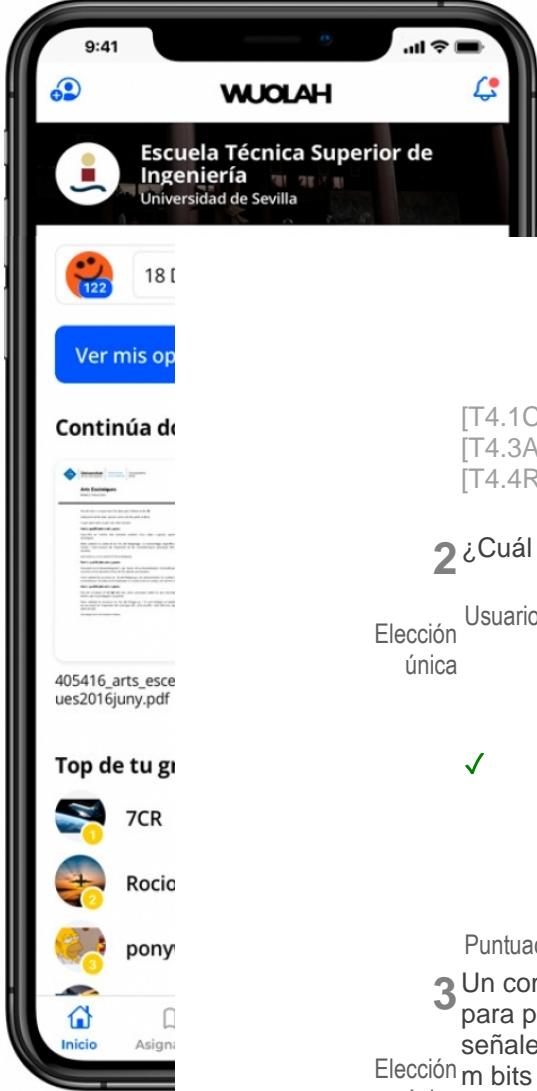
1 [T4.4]

La segmentación de cauce...

Elección única
Usuario Profesores

- a) permite ejecutar varias instrucciones concurrentemente
- b) acelera la ejecución de un programa
- c) provoca riesgos debido a datos
- ✓ d) todas las respuestas son ciertas

Puntuación: **1,00**



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

Continúa d
Elección única
Usuario Profesores
[T4.1ConSeg]
[T4.3Aceler]
[T4.4Riesgs]

2 ¿Cuál de las siguientes afirmaciones es incorrecta?

- Elección única Usuario Profesores
- ✓ •
- a) El repertorio de instrucciones es el conjunto de operaciones que es capaz de interpretar la unidad de control.
 - b) Los operandos siempre están almacenados en memoria.
 - c) El repertorio de instrucciones debe ser capaz de realizar una tarea en un tiempo finito.
 - d) El modo de direccionamiento permite determinar un operando o la ubicación del operando.

Puntuación: 1,00

3 Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?

- Elección única Usuario Profesores
- X •
- a) $k + n \cdot 2^m$
 - b) $k + n^m$
 - c) $k + n \cdot (2^m - 1)$
 - d) Ninguno de los anteriores

Puntuación: -0,33

4 [T2.1.2]
La instrucción leave hace exactamente lo mismo que

- Elección única Usuario Profesores
- ✓ •
- a) mov %esp, %ebp; pop %esp
 - b) pop %eip
 - c) ret
 - d) mov %ebp, %esp; pop %ebp

Puntuación: 1,00

[T2.1.2Lngjes]
[T2.3.1Marcop]
[E15FebPra11]

5 [T2.4.3]

Al traducir la sentencia C r->i = val; gcc genera el código ASM movl %edx, 12(%eax). Se deduce que

Elección Usuario Profesores
única



- i es un entero que vale 12
- a) después de la asignación, i vale lo que había en %edx (que es val)
 - b) el desplazamiento de i en *r es 12 y en concreto r está en %eax
 - c) val es un entero que vale 12 val está en %edx
 - r es un puntero que apunta a la posición de memoria 12
 - d) r es puntero a struct y seguramente valdrá 0x0804XXXX en ejecutables x86

Puntuación: **1,00**

[T2.4.3Struct]

[E15SepPra13]

6 [T6.3]

Un sistema basado en un microprocesador con un bus de datos de n bits y un bus de direcciones de 16 bits direcciona la memoria por palabras de n bits y dispone de una memoria SRAM formada por dos módulos de 16 K x n cada uno. ¿Qué porcentaje del mapa de memoria está ocupado por la SRAM?

Usuario Profesores



- 50%
- a) 16bits => 64Kpal
 $16+16 \text{ Kpal} = 32\text{Kpal} = 50\% \text{ 64Kpal}$
- b) 25%
- c) 100%
- d) 12,5%

Puntuación: **-0,33**

[T1.3EstBus]

[T6.3Diseño]

[E17FebTeo27]

7 [T5.1]

La conexión entre un dispositivo de E/S y el procesador mediante bus:

Elección Usuario Profesores
única



- a) Requiere mucha circuitería
- b) Permite conectar en paralelo varios dispositivos
- c) Requiere multiplexores y demultiplexores para las señales de datos
- d) Es difícil de expandir

Puntuación: **-0,33**

[T1.3EstBus]
[T5.1FunE/S]
[E17JulTeo19]

8 Una memoria que está estructurada en palabras de 8 bits tiene una capacidad de 32 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?

Elección
única Usuario Profesores

- ✓ • a) 12
b) 8
c) 4
d) 32

Puntuación: **1,00**

9 El espacio direccionable de memoria de un computador depende del diseño del:

Elección
única Usuario Profesores

- a) Bus de direcciones
b) Bus de datos
c) Ninguna de las otras es correcta
X d) a) y b) son correctas

Puntuación: **-0,33**

10 [T6.2]

¿Cuál de las siguientes afirmaciones es falsa?

Elección
única Usuario Profesores

- ✓ • La lectura de un bit de la matriz de almacenamiento de una memoria DRAM proporciona una señal mucho más débil que la suministrada por los inversores de una celda de memoria SRAM.
a) Una celda DRAM sólo necesita un transistor y un condensador.
c) Las memorias DRAM son en general mucho más rápidas que las SRAM
Las memorias DRAM presentan generalmente una d) capacidad de almacenamiento mucho mayor que las SRAM.

Puntuación: **1,00**

[T6.2RAMROM]

11 La operación aritmética calculada por el programa

Elección
única

```
mov r1,#5
    mov r2,#3
    mov r3,#7
    mov r4,#8
    mul r5,r2,r3
```

add r6,r1,r5
sub r7,r6,r4

es:

Usuario Profesores

- ✓ • a) $8 - 5 + (3 \times 7)$
b) $5 + (3 \times 7) - 8$
c) $8 - (3 \times 7) + 5$
d) $(3 \times 7) + 8 - 5$

Puntuación: 1,00

12 ¿Cuál es la diferencia entre las instrucciones subl y cmpl?

Elección única Usuario Profesores

- ✓ • a) subl tiene en cuenta el acarreo de entrada y cmpl no
b) subl no afecta a los indicadores de estado y cmpl sí
c) subl almacena el resultado sobre escribiendo uno de los operandos y cmpl no
d) subl realiza una resta y cmpl realiza una suma

Puntuación: 1,00

13 [T1.1]

Son funciones de la unidad de control:

Elección única Usuario Profesores

- X • a) la codificación de las instrucciones máquina
b) la lectura de memoria principal de la instrucción apuntada por el μ PC
c) el secuenciamiento de las instrucciones máquina
d) todas las respuestas son ciertas

Puntuación: -0,33

[T1.1UniFun]

[T3.1CamDat]

[E17JulTeo07]

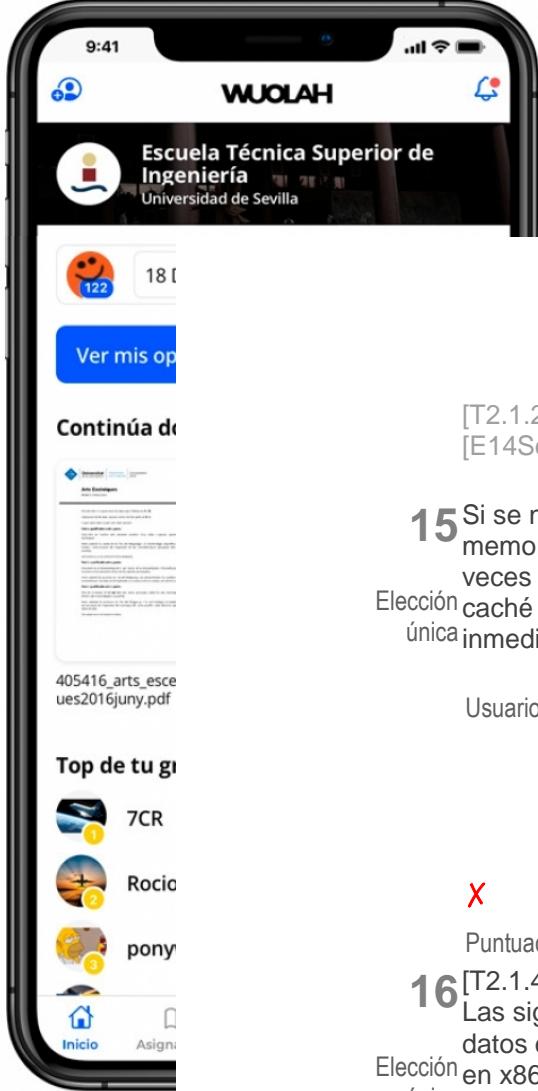
14 [T2.1.2]

Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:

Elección única Usuario Profesores

- ✓ • a) compilador
b) simulador
c) ensamblador
d) intérprete

Puntuación: 1,00



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- 15** [T2.1.2Lngjes]
[E14SepTeo14]
- Si se necesitan 60 ns para escribir una palabra de datos de caché en memoria principal y cada bloque de caché tiene 8 palabras, ¿cuántas veces seguidas se tiene que escribir en un mismo bloque para que una caché de postescritura sea más eficiente que una de escritura única inmediata?

Usuario Profesores

- a) Más de 8 veces.
- b) Depende de la tasa de aciertos.
- c) La caché de postescritura no puede ser más eficiente que la de escritura inmediata.
- d) La caché de postescritura siempre será más eficiente que la de escritura inmediata.

Puntuación: **-0,33**

- 16** [T2.1.4]
Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA32 como en x86-64. Sólo una de ellas es FALSA. ¿Cuál?

Usuario Profesores

- ✓ • a) El tamaño de un double es 8 bytes
- ✓ • b) El tamaño de un puntero es 4 bytes
- ✓ • c) El tamaño de un short es 2 bytes
- ✓ • d) El tamaño de un int es 4 bytes

Puntuación: **1,00**

[T2.1.4x86-64]
[E12SepTeo08]

- 17** ¿En qué tipo de memoria virtual es un problema la fragmentación externa?

Elección única
Usuario Profesores

- a) Memoria paginada
- b) Memoria segmentada
- c) Memoria con segmentación paginada
- d) En ninguno de ellos

Puntuación: **0,00**

- 18** [T4.4]
Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?

Elección única
Usuario Profesores

- ✓ •
- a) el retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca
 - b) el retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple
 - c) el retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de salto
 - d) el retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)

Puntuación: 1,00

[T4.4Riesgs]
[E15SepTeo19]

19 ¿Cuál de las siguientes funciones no corresponde a un módulo de E/S?

Elección única Usuario Profesores

✓ •

- a) Almacenamiento de programas
- b) Comunicación con el dispositivo
- c) Almacenamiento temporal de datos
- d) Comunicación con el microprocesador

Puntuación: 1,00

20 [T2.3.1] Sobre el direccionamiento relativo a contador de programa:

Elección única Usuario Profesores

✓ •

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos y llamadas a subrutinas reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está ejecutando.
- d) Todas las respuestas son ciertas.

Puntuación: 1,00

[T2.2.4SalCon]
[T2.3.1MarcoP]

21 [T2.1.2] En IA32 la pila es:

Elección única Usuario Profesores

X

•

- a) una dirección de memoria de 32 bits almacenada en el contador de programa
- b) un registro de 32 bits en el microprocesador
- c) un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa

d) un registro de 16 bits en el microprocesador

Puntuación: **-0,33**

[T2.1.2Lngjes]

[T2.3.1MarcoP]

[E14SepTeo15]

[E16FebTeo19]

22 [T2.4.2]

En el siguiente código, ¿qué reordenamiento de los bucles muestra mejor localidad?

Elección

única // X, Y, Z ctes #define previo
int a[X][Y][Z]
int i, j, k, sum = 0;
for (i = 0; i < Y; i++)
 for (j = 0; j < Z; j++)
 for (k = 0; k < X; k++)
 sum += a[k][i][j];

Usuario Profesores

- ✓ • a) k externo, i central, j interno
b) j externo, k central, i interno
c) El orden de los bucles no afecta a la localidad
d) i externo, j central, k interno (el orden en que están ahora)

Puntuación: **1,00**

[T2.4.2Arrays]

[T6.1ConLoc]

[E14FebTeo04]

23 [T5.1]

Respecto a si un computador dispone de E/S independiente (separada) o usa E/S mapeada a memoria:

Elección

única Usuario Profesores

- Si el encapsulado del procesador no dispone de
a) patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada
Si el repertorio del procesador tiene instrucciones
b) del tipo IN y OUT, es que el computador dispone de E/S separada
Si el repertorio del procesador tiene instrucciones
c) del tipo LOAD y STORE, el computador sólo dispone de E/S mapeada a memoria
Si el encapsulado (chip) del procesador tiene patilla
d) (pin) IO/M# (o patillas equivalentes), eso evidencia que el computador usa E/S mapeada a memoria

Puntuación: **1,00**

[T5.1FunE/S]
[E15FebTeo16]

24 ¿Cuántos puertos de E/S permite manejar la interfaz de periféricos programable 8255?

Elección
única

X

•

- a) 3 puertos de 12 bits
- b) 2 puertos de 16 bits y 1 puerto de 8 bits
- c) 2 puertos de 8 bits y 2 puertos de 4 bits
- d) Todas las respuestas anteriores son falsas

Puntuación: -0,33

25 [T2.2.1]

¿Cuál de las siguientes instrucciones x86 se puede usar para sumar dos registros y guardar el resultado sin sobrescribir ninguno de los registros originales?

Elección
única

Usuario Profesores

✓

•

- a) mov
- b) lea
- c) add
- d) Ninguna de ellas

Puntuación: 1,00

[T2.1.3ConASM]

[T2.2.1ModDir]

[T2.2.2OpArit]

[E12FebTeo06]

26 Al diseñar el formato de instrucción:

Elección
única

Usuario Profesores

•

- a) el número de formatos de instrucción diferentes no afecta a la complejidad de la UC.
- b) se suele omitir el campo que indica la siguiente instrucción (la siguiente a ejecutar es la siguiente en memoria, salvo en caso de salto).
- c) sólo hay que saber el tipo de operación a realizar y los operandos necesarios.
- d) hay que indicar explícitamente todos los operandos y destinos.

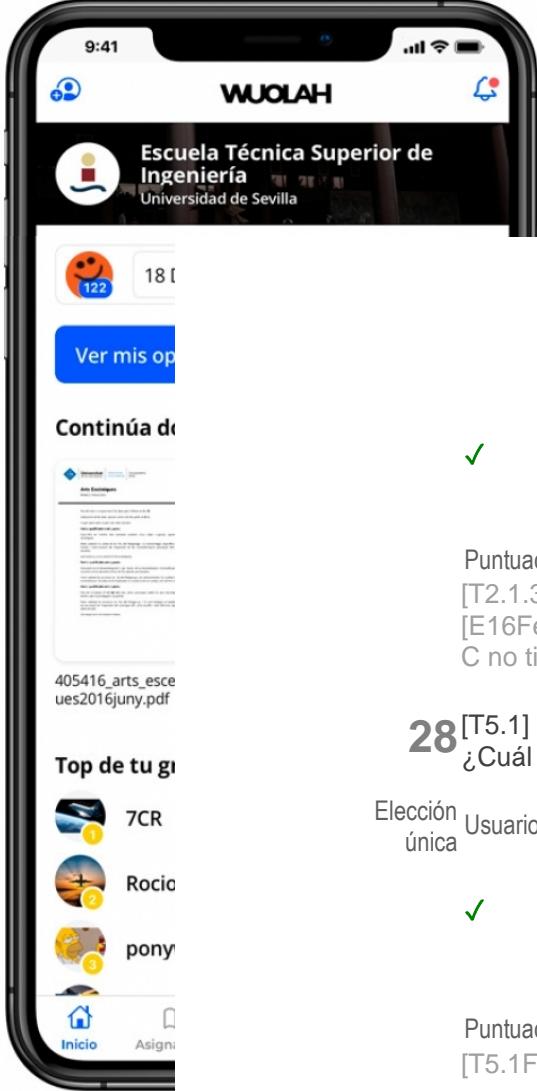
Puntuación: 0,00

27 [T2.1.3]

¿Cuál de los siguientes lenguajes no permite el paso de parámetros por referencia?

Elección
única

Usuario Profesores



Descarga la APP de Wuolah.
Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- a) C++
- b) C
- c) FORTRAN
- d) Pascal

Puntuación: 1,00

[T2.1.3ConASM]
[E16FebTeo13]

C no tiene "referencias", por eso se pasan punteros a la función swap()

28 [T5.1]

¿Cuál de las siguientes afirmaciones es cierta?

Elección
única

Usuario Profesores

- ✓
-

- a) La E/S en memoria emplea la patilla IO/M#
- b) En E/S independiente, las instrucciones de acceso a memoria suelen ser más largas que las de E/S
- c) La E/S en memoria facilita la protección
- d) Ninguna de las anteriores es cierta

Puntuación: 1,00

[T5.1FunE/S]

29

Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?

Elección
única

Usuario Profesores

- X
-

- a) 8192 milisegundos
- b) 32,768 segundos
- c) 7,8125 microsegundos
- d) 31,25 microsegundos

Puntuación: -0,33

30

El resultado de desplazar aritméticamente dos posiciones hacia la derecha el número de 8 bits en complemento a dos –32 es:

Elección
única

Usuario Profesores

-

- a) 56
- b) -128
- c) -8
- d) Ninguno de los resultados anteriores es correcto

Puntuación: 0,00

LAS PREGUNTAS MÁS INTERASANTES

11 [P2.1]

En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo único valor entero que repetido en toda la lista causaría acarreo con 32 bits (sin signo)? Se usa notación decimal y espacios como separadores de millares/millones/etc.

Usuario Profesores

X

- a) 10 000 000
- b) 100 000 000
No llega, 100 millones < 128M
- c) 1 000 000 000
Se pasa, 1000 millones >> 128M
- d) 10 000 000 000

Puntuación: **-0,33**

[P2.1SumUns]

[E17FebPra08]

32bit sin signo llega a $2^{32}-1 = 4G-1 \approx 4\ 000\ 000\ 000$.

Habría acarreo si llegáramos a 4G. Con 32 elementos iguales, tendrían que valer $4G/32 = 1G/8 = 128M$.

24 [T2.3.1]

En la secuencia de programa siguiente:

Elección única
804854e:e8 3d 06 00 00 call 8048b90 <main>
8048553:50 pushl %eax

¿cuál es el valor que introduce en la pila la instrucción call?

Usuario Profesores

- a) 0x8048b90
- b) 0x804854f
- c) 0x8048553
- d) 0x804854e

Puntuación: **0,00**

[T2.3.1MarcoP]

[E16FebTeo14]

2 [P2.2]

En la práctica "media" se pide sumar una lista de enteros *con* signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en los *dos* primeros elementos de la

Elección lista causaría overflow con 32 bits al realizar la suma de *esos dos* primeros elementos de la lista?

Usuario Profesores

- a) 0x0400 0000
suma 0x0800 0000 sin problema
0x8000 0000
- b) es negativo, el enunciado pide menor valor positivo
- c) 0x0800 0000
suma 0x1000 0000 sin problema
- d) 0x4000 0000
0x8000 0000 sería negativo en 32bit

✓ • Puntuación: 1,00

[P2.2SumSgn]
[E17JulPra09]

3 [P2T]

El switch de gcc para que únicamente compile de lenguaje C a ensamblador, y no realice ningún paso adicional (ensamblar, enlazar, etc), es...

Elección

única

Usuario Profesores

- a) -g
para incorporar info depuración
- b) -c
de C/asm a objeto, .c/.s→.o
- c) -o
para nombrar el ejecutable
- d) -S

✓ • Puntuación: 1,00

[P2Tutorial]
[E16SepPra01]

5 [P2T]

La siguiente línea en la sección de datos de un programa en ensamblador de IA32
result: .int 0,0

Usuario Profesores

- a) Reserva espacio para un único entero, inicializado a 0,0
Reserva espacio para un único entero, b) inicializado a 0, en la posición de memoria 0

- ✓ •
- c) Reserva espacio para un entero, inicializado a 0, seguido de un dato de tamaño indefinido, también inicializado a 0
 - d) Reserva espacio para dos enteros, inicializados ambos a 0

Puntuación: **1,00**

[P2Tutorial]
[E17FebPra04]

7 [P4T]

Elección única En la práctica de la bomba necesitamos estudiar el código máquina de la bomba del compañero. A veces dicho código no se visualiza directamente en el depurador ddd, y algunas de las técnicas que se pueden probar para conseguir visualizarlo son... (marcar la opción *falsa*)

Usuario Profesores

- X •
- a) comprobar que está activado el panel View
→ Machine Code Window
 - b) asegurarse de que se ha escrito correctamente el nombre del ejecutable
 - c) escribir info line main en el panel de línea de comandos gdb
recompilar con información de depuración, por si se nos había olvidado, ya que sin `-g` el ejecutable no contiene información de depuración
sí, claro, como que nos van a pasar el código fuente... :-)
 - d)

Puntuación: **-0,33**

[P4Tutorial]
[E17JulPra14]

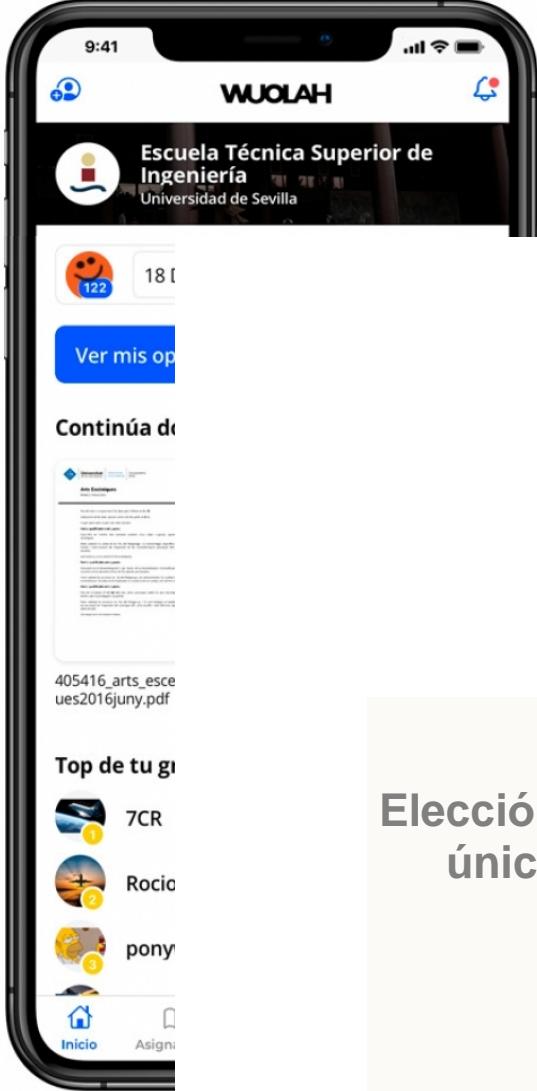
9 [P2.2]

Elección única En la práctica “media” se pide sumar una lista de 32 enteros *con* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista causaría overflow con 32bits?

Usuario Profesores

0xf000 0000

- a) se pierde el signo con `<<4`, mucho más con `<<5`
- b) 0xffff ffff
eso es -1, y $-1 \times 32 == -32$ sin problemas
- c) 0xfbff ffff



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

en cuanto sea algo más grande que
0xfc00 0000 sale overflow
0xfc00 0000

- d) $nx32 == n << 5$ y quedaría 0x8000 0000 !!!
justo para que no haya overflow

Puntuación: 1,00

[P2.2SumSgn]
[E15SepPra16]

6 [T6.3]

Elección única

Un sistema basado en un microprocesador con un bus de datos de n bits y un bus de direcciones de 16 bits direcciona la memoria por palabras de n bits y dispone de una memoria SRAM formada por dos módulos de 16 K x n cada uno. ¿Qué porcentaje del mapa de memoria está ocupado por la SRAM?

Usuario Profesores

- a) 50%
16bits => 64Kpal
16+16 Kpal = 32Kpal = 50% 64Kpal
- b) 25%
- c) 100%
- d) 12,5%

X Puntuación: -0,33

[T1.3EstBus]
[T6.3Diseño]
[E17FebTeo27]

9 El espacio direccionable de memoria de un computador depende del diseño del:

Elección única

Usuario Profesores

- a) Bus de direcciones
- b) Bus de datos
- c) Ninguna de las otras es correcta
- d) a) y b) son correctas

X Puntuación: -0,33

15 Si se necesitan 60 ns para escribir una palabra de datos de caché en memoria principal y cada bloque de caché tiene 8 palabras, ¿cuántas veces seguidas se tiene que escribir en un mismo bloque para que una caché de postescritura sea más eficiente que una de escritura inmediata?

Elección única

Usuario Profesores

		<ul style="list-style-type: none"> • a) Más de 8 veces. • b) Depende de la tasa de aciertos. • c) La caché de postescritura no puede ser más eficiente que la de escritura inmediata. • d) La caché de postescritura siempre será más eficiente que la de escritura inmediata. <p>X</p>
		Puntuación: -0,33
18	[T4.4] Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?	
Elección única	Usuario Profesores	<p>✓</p> <ul style="list-style-type: none"> • a) el retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca • b) el retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple • c) el retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de salto • d) el retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)
		Puntuación: 1,00
	[T4.4Riesgs] [E15SepTeo19]	
21	[T2.1.2] En IA32 la pila es:	
Elección única	Usuario Profesores	<p>X</p> <ul style="list-style-type: none"> • a) una dirección de memoria de 32 bits almacenada en el contador de programa • b) un registro de 32 bits en el microprocesador • c) un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa

- d) un registro de 16 bits en el microprocesador

Puntuación: **-0,33**

[T2.1.2Lngjes]
[T2.3.1MarcoP]
[E14SepTeo15]
[E16FebTeo19]

22 [T2.4.2]

En el siguiente código, ¿qué reordenamiento de los bucles muestra mejor localidad?

Elección única

```
// X, Y, Z ctes #define previo
int a[X][Y][Z]
int i, j, k, sum = 0;
for (i = 0; i < Y; i++)
    for (j = 0; j < Z; j++)
        for (k = 0; k < X; k++)
            sum += a[k][i][j];
```

Usuario Profesores

- ✓ • a) k externo, i central, j interno
b) j externo, k central, i interno
c) El orden de los bucles no afecta a la localidad
d) i externo, j central, k interno (el orden en que están ahora)

Puntuación: **1,00**

[T2.4.2Arrays]
[T6.1ConLoc]
[E14FebTeo04]

26

Al diseñar el formato de instrucción:

Elección única

Usuario Profesores

- a) el número de formatos de instrucción diferentes no afecta a la complejidad de la UC.
- b) se suele omitir el campo que indica la siguiente instrucción (la siguiente a ejecutar es la siguiente en memoria, salvo en caso de salto).
- c) sólo hay que saber el tipo de operación a realizar y los operandos necesarios.

- d) hay que indicar explícitamente todos los operandos y destinos.

Puntuación: **0,00**

27 [T2.1.3]

¿Cuál de los siguientes lenguajes no permite el paso de parámetros por referencia?

Elección única

Usuario Profesores

- ✓ • a) C++
- b) C
- c) FORTRAN
- d) Pascal

Puntuación: **1,00**

[T2.1.3ConASM]

[E16FebTeo13]

C no tiene "referencias", por eso se pasan punteros a la función swap()

EDEN

El direccionamiento relativo a registro base utiliza...

Usuar Profesor

io es

- a) dos registros.
- b) un registro.
- c) dos desplazamientos contenidos en la propia instrucción.
- ✓ • d) un registro y un desplazamiento.

Puntuación: **1,00**

Una instrucción de salto si menor, para números positivos sin signo, tiene que comprobar el valor de:

Usuar Profesor

io es

- a) los bits de signo y desbordamiento
- b) el bit de signo
- ✗ • c) el bit de cero
- d) el bit de acarreo

Puntuación: **-0,33**

La instrucción setg %al:

U Pro
su fes
ari ore
o s

- a Pone siempre AL a 1.
)
- ✓ • b Pone AL a 1 en algunos casos.
)
- c No cambia el contenido de AL
)
- d Complementa AL si el resultado
) de la comparación anterior es A >
B.

Puntuación: **1,00**

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

Usuar Profesor

io es

- a) RET
- b) CALL dir
- c) JNE dir
- d) JMP dir

Puntuación: **1,00**

El lenguaje máquina es...

Usuar Profesor

io es

- a) fácilmente legible por el programador.
- b) portable entre arquitecturas.
- c) difícil de codificar manualmente.
- d) una alternativa razonable al uso del lenguaje ensamblador.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la segmentación de cauce?

Usuar Profesor

io es

- a) primera
- b) segund
a
- c) tercera
- d) cuarta

Puntuación: **1,00**

[T1.5]

¿En qué generación, dentro de la historia de los computadores digitales, se alcanzaron tiempos de conmutación del orden de nanosegundos?

Usuar Profesor

io es

- a) primera

- b) segunda
- c) tercera
- d) cuarta

Puntuación: **1,00**

En IA32, el registro contador de programa se denomina:

Usuar Profesor

io es

- a) PC
- b) RIP
- c) PC
R
- d) EIP

Puntuación: **-0,33**

En cdecl/x86, ¿cuál de los siguientes registros tiene que ser guardado por la función llamada si es alterado por ésta?

- a) eax
- b) ecx
- c) ebx
- d) edx

¿Cuál de las siguientes listas está correctamente ordenada temporalmente?

- a) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
- b) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
- c) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
- d) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX

Si el contenido de EBP es 0x13000, ¿a qué dirección se hace referencia con la instrucción INC –0x80(%EBP)?

- ✓ a) 0x13080
- b) 0x13000
- c) 0x80
- d) 0x12F80

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

- ✗ a) 53000h
- b) **4C000h**
- c) 15025h
- d) 5C000h

7 [T1.2]

El modo de direccionamiento en el que se especifica un registro y una dirección de memoria cuyo contenido se suma al contenido del registro base para obtener la dirección efectiva, se conoce como:

ón Us Prof
ún ua eso
ic río res

- a a base con
 -) desplazamiento
- b directo o absoluto
 -)
- ✗ c indirecto a registro
 -) través de memoria
- d ninguno de los
 -) anteriores

Puntuación: **-0,33**

8 [T1.5]

Elección única

Usuar Profesor

io es

- a) primera
- b) segund
- a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

Sobre el ensamblador:

Usuar Profesor

io es

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
- ✓ • d) Todas las respuestas son ciertas.

Suponiendo que todos los registros inicialmente contienen el valor 1 y que el destino es el primer registro, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4  
mov r2, #3  
add r3, r1, r1
```

sub r1, r3, r2

mul r3, r1, r1

Usuar Profesor

io es

- a) 3
- 6
- b) 6
- ✓ • c) 5
- d) 2
- 0

Puntuación: **1,00**

[T2.2.3]

En los casos concretos indicados para las siguientes instrucciones IA32, ¿cuál no funciona como instrucción de transferencia?

Usuar Profesor

io es

- ✓ • a) cmpl 0x08048040, %eax
- b) movl \$0x15, %eax
- c) pushl %eax
- d) leal variable, %eax

Puntuación: **1,00**

¿Qué medida de prestaciones es la más fiable de todas las posibles?

Usuar Profesor

io es

- a) MIPS
- ✗ b) MIPS equivalentes
- c) ninguna de las otras respuestas es correcta
- d) MFLOPS

Puntuación: **-0,33**

En un microprocesador de 4 bits, una operación en la que el bit 0 de un registro se copia en el acarreo, después el bit 1 se copia en el bit 0, después el bit 2 se copia en el bit 1, y por último el bit 3 se copia en el bit 2, es:

Usuar Profesor

io es

- a) Un desplazamiento lógico a la derecha.
- b) Un desplazamiento aritmético a la derecha.
- c) Una rotación a la derecha.
- d) Una rotación a la derecha a través de acarreo.

Puntuación: **-0,33**

El ancho de palabra de una memoria corresponde a:

Usuar Profesor

io es

- a) La longitud del registro de datos de la memoria.
- b) El número de posiciones que la componen.
- c) El número que identifica únicamente cada posición de la memoria.
- d) La longitud del registro de direcciones de la memoria.

Puntuación: **-0,33**

Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro

Usuar Profesor

io es

- a) De direcciones
- b) Puntero de pila
- c) Contador de programa
- d) De instrucción

Puntuación: **1,00**

El objetivo de un diseño CISC es...

Usuar Profesor

io es

- a) disminuir el tamaño medio de instrucción.
- b) disminuir la frecuencia de reloj.
- ✓ • c) disminuir el número de instrucciones a ejecutar por un programa.
- d) disminuir el número medio de ciclos por instrucción.

Puntuación: **1,00**

En la captación de la instrucción:

Usuar Profesor

io es

- a) en MBR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- b) en MAR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- ✓ • c) en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción.
- d) en MBR indicamos la dirección donde está la instrucción y en MAR recogemos la instrucción.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparecen las memorias de semiconductores?

Usuar Profesor

io es

- a) Tercera generación.
- b) Cuarta generación.
- ✗ c) Segunda generación.
- d) Quinta generación.

Puntuación: **-0,33**

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

Usuar Profesor

io es

- a) BP

- b) SS * 10h +
SP
- X** c) SP
- d) SS * 10h +
BP

Puntuación: **-0,33**

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

Usuar Profesor

io es

- ✓**
- a) directo a registro
 - b) a registro inmediato
 - c) a registro literal
 - d) ninguno de los anteriores

Puntuación: **1,00**

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

Usuar Profesor

io es

- ✓**
- a) directo a registro
 - b) a registro inmediato
 - c) a registro literal
 - d) ninguno de los anteriores

Puntuación: **1,00**

Si AX = 0xFA50 y ejecutamos AND \$0xFF, %AX

Usuar Profesor

io es

- X**
- a) El registro AH se pone a FF
 - b) El registro AH se pone a 0
 - c) El registro AL se pone a FF
 - d) El registro AL se pone a 0

Puntuación: **-0,33**

En una máquina con 32 registros direccionables e instrucciones de 16 bits:

Usuar Profesor

io es

X

- a) no se pueden codificar a la vez 64 instrucciones de 0 direcciones, 63 instrucciones de dos registros y 16 instrucciones de 1 registro.
- b) no se pueden codificar a la vez 63 instrucciones de dos registros, una instrucción de 0 direcciones y 32 instrucciones de 1 registro.
- c) no se pueden codificar a la vez 64 instrucciones de 1 registro, 32 instrucciones de dos registros y 32 instrucciones de 0 direcciones.
- d) no se pueden codificar a la vez 62 instrucciones de dos registros, 32 instrucciones de 1 registro y 64 instrucciones de 0 direcciones.

Puntuación: **-0,33**

7

¿Qué tipos de instrucciones se emplean más en una arquitectura de registros de propósito general?

Elecció
n única

Usua
rio Profes
ores

X

- a) De desplazamiento y rotación.
- b) Aritmético-lógicas.
- c) De transferencia de datos.
- d) De transferencia de control.

Puntuación: **-0,33**

8

Elecció n únic a Si d es un desplazamiento, r un registro índice e i una constante apropiada, el modo de direccionamiento indexado con postautodecremento realiza...

Usua rio
n
únic a
Profes ores



- a) dirección efectiva = $r + d$; $r = r - i$
- b) dirección efectiva = $r - i$; $r = r - d$
- c) $r = r - i$; dirección efectiva = $r - d$
- d) $r = r + i$; dirección efectiva = $r - d$

Puntuación: 1,00

En el direccionamiento inmediato:

Usuar Profesor

io es



- a) el tamaño (rango de valores) de la constante está limitado.
- b) el operando es una constante de tamaño arbitrario contenida en la misma instrucción.
- c) el código de operación indica el operando afectado.
- d) una vez captada y decodificada la instrucción, se accede a memoria para obtener el valor del operando.

Puntuación: 1,00

En el RISC-I, una ventana de registros contiene:

Usuar Profesor

io es



- a) registros para recibir parámetros del procedimiento llamador.
- b) registros para almacenar variables locales.
- c) registros para enviar parámetros a procedimientos.
- d) todas las respuestas son ciertas.

Puntuación: -0,33

Si queremos almacenar la palabra de 16 bits 0x9660 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x1000 como:

Usuar Profesor

io es

- a) M[0x1000]=0x06 y
M[0x1001]=0x69
- b) M[0x1000]=0x69 y
M[0x1001]=0x06
- ✓ • c) M[0x1000]=0x60 y
M[0x1001]=0x96
- d) M[0x1000]=0x96 y
M[0x1001]=0x60

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?

Usuar Profesor

io es

- a) primera
- b) segund
a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

GCC/Linux IA32 resuelve el ajuste de marco de pila al comenzar un procedimiento mediante las instrucciones:

Usuar Profesor

io es

- a) pushl %ebp movl %esp,
%ebp
- b) movl %esp, %ebp popl %esp
- ✗ c) pushl %esp movl %ebp,
%esp
- d) movl %ebp, %esp popl %ebp

Puntuación: **-0,33**

¿En qué orden debería ejecutarse en una máquina de tipo pila la operación aritmética ($a+b/c-d$)?

Usuar Profesor

io es

a) $a + b / c -$

d

b) $a b + c d -$

/



• c) $a b c / + d$

-

d) $a b + / c d$

-

Puntuación: **1,00**

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

Usuar Profesor

io es

a) registro-registro

b) registro-memoria



• c) memoria-memoria

d) ninguna de las anteriores es
cierta

Puntuación: **1,00**

¿Cuál de las siguientes parejas de microprocesadores representa mejor los conceptos RISC?

Usuar Profesor

io es



• a) MIPS, SPARC

b) PA-RISC,
PowerPC

c) Itanium, Alpha

d) Pentium, Athlon

Puntuación: **1,00**

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

Usuar Profesor

io es

- a) SS * 10h +
 BP
- b) BP
- c) SS * 10h +
 SP
- X d) SP

Puntuación: **-0,33**

9 [T1.5]

Elecció
n única

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?

Usuar Profesor

io es

- a) primera
- b) segunda
- ✓ c) tercera
- d) cuarta

Puntuación: **1,00**

2

[T1.1]

Elección
única

Usu Profe
ario sores

- ✓ • a de transferencia de datos con
) memoria
- b aritmético-lógicas
)
- c de transferencia de datos
) entre registros

d de desplazamiento y rotación
)

Puntuación: **1,00**

Una máquina superescalar es aquella que:

Usuar Profesor

io es

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- ✓ • c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
- d) ninguna respuesta de las anteriores es correcta.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la segmentación de cauce?

Usuar Profesor

io es

- a) primera
- b) segund
- a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

Si el registro EAX contiene X, La secuencia de instrucciones siguiente:

cmpl \$6, %eax

jae Destino

salta a la etiqueta Destino sólo si:

Usuar Profesor

io es

- ✓ • a) $X < 0 \text{ || } X \geq 6$
- b) $X \leq 6$
- c) $X \geq 0 \text{ && } X \leq 6$
- d) $X > 6$

Puntuación: **1,00**

El lenguaje máquina es...

Usuar Profesor

io es

- ✓ • a) difícil de codificar manualmente.
b) portable entre arquitecturas.
c) fácilmente legible por el programador.
d) una alternativa razonable al uso del lenguaje ensamblador.

Puntuación: **1,00**

Si usamos una estructura de bus con DMA:

Usuar Profesor

io es

- a) al bus del sistema sólo se conecta la CPU y la MP, ya que el DMA se conecta directamente a MP para realizar las transferencias de datos.
- ✗ b) podemos prescindir de controladores de E/S ya que el controlador de DMA se ocupa de controlar las transferencias hacia/desde los periféricos.
- c) la velocidad de este controlador establece la velocidad del bus del sistema.
- d) la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.

Puntuación: **-0,33**

Sobre el direccionamiento relativo al contador de programa:

Usuar Profesor

io es

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está ejecutando.
- ✓ • d) Todas las respuestas son ciertas.

Puntuación: **1,00**

Si %eax contiene x, ¿cuál de las siguientes instrucciones calcula x*5?

Usuar Profesor

io es

- a) leal 3(%eax,2),%edx
- b) leal 3(%eax,%eax,2),%edx
- c) leal 4(%eax,%eax),%edx
- d) leal (%eax,%eax,4),%edx

Puntuación: **1,00**

Si queremos almacenar la palabra de 16 bits 8965h en memoria según "big-endian", quedará almacenada a partir de la posición 1000h como:

Usuar Profesor

io es

- a) en el byte 1000h se guarda A6h y en el 1001h 91h
- b) en el byte 1000h se guarda 91h y en el 1001h A6h
- c) en el byte 1000h se guarda 89h y en el 1001h 65h
- d) en el byte 1000h se guarda 65h y en el 1001h 89h

Puntuación: **1,00**

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

Usuar Profesor

io es

- a) 15025h
- b) 4C000 h
- c) 5C000 h
- d) 53000h

Puntuación: **1,00**

Sobre el ensamblador:

Usuar Profesor

io es

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
- ✓ • d) Todas las respuestas son ciertas.

Puntuación: **1,00**

Un procesador emplea codificación en bloque del código de operación. Existen 123 instrucciones que tienen una longitud de 16 bits. ¿A cuántas direcciones de memoria pueden acceder como máximo si todas emplean una estructura “código de operación + dirección de memoria”?

Usuar Profesor

io es

- a) 128.
- ✗ b) 256.
- c) 1024
- .
- d) 512.

Puntuación: **-0,33**

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

Usuar Profesor

io es

- a) registro-registro
- b) registro-memoria
- ✓ • c) memoria-memoria
- d) ninguna de las anteriores es cierta

Puntuación: **1,00**

Para direccionar una memoria de 1 G palabras de 32 bits cada una, que se direcciona byte a byte, se necesitarán:

Usuar Profesor

io es

- a) 33
bits
- b) 21
bits
- c) 32
bits
- d) 31
bits

Puntuación: **-0,33**

[T1.5]

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria virtual?

Usuar Profesor

io es

- a) primera
- b) segund
a
- c) tercera
- d) cuarta

Puntuación: **1,00**

En el RISC-I, una ventana de registros contiene, entre otros registros,...

Usuar Profesor

io es

- a) registros para recibir parámetros del procedimiento llamador
- b) registros para almacenar matrices de enteros
- c) registros para enviar parámetros a otros procesos
- d) todas las respuestas son falsas

Puntuación: **1,00**

En el arbitraje de un bus...

Usuar Profesor

io es

- a) los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
- b) si hay un único dispositivo pasivo, siempre funciona como esclavo
- c) si hay varios dispositivos activos, siempre funcionan como maestros
- X d) todas las respuestas anteriores son ciertas

Puntuación: **-0,33**

¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?

Usuar Profesor

io es

- a) posición 0: FF pos.1:FF pos.2: FF pos.3:
80
- ✓ • b) 0:80 1:FF 2:FF 3:FF
- c) 0:00 1:01 2: 00 3:80
- d) Ninguna de las anteriores

Puntuación: **1,00**

Si A=FF0Fh y B=0004h, el resultado de desplazar A a la derecha aritméticamente B veces es:

Usuar Profesor

io es

- a FFF0h
)
- X b 0FF0h
)
- c F0FFh
)
- d F0F0h
)

Puntuación: **-0,33**

Suponiendo que todos los registros inicialmente contienen el valor 0, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4  
mov r2, #3  
add r3, r1, r1  
sub r1, r3, r2  
mul r3, r1, r1
```

Usuar Profesor

io es

- a) 4
- b) 5
- c) 2
- 5
- d) 0

Puntuación: **1,00**

Un modo de vídeo de 1680 x 1050 píxeles y 4 canales de color por píxel (R, G, B, alpha) y 256 niveles de intensidad por canal (0 a 255), ocupa aproximadamente una memoria de:

Usuar Profesor

io es

- a) 1,7 MB
- b) 7 MB
- c) 54 MB
- d) 1,7 GB

Puntuación: **1,00**

Suponiendo que todos los registros inicialmente contienen el valor 0, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4  
mov r2, #3  
add r3, r1, r1  
sub r1, r3, r2  
mul r3, r1, r1
```

Usuar Profesor

io es

- a) 4
- b) 5
- c) 25
- d) 0

Puntuación: **1,00**

Una máquina superescalar es aquella que:

Usuar Profesor

io es

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
- X d) ninguna respuesta de las anteriores es correcta.

Puntuación: **-0,33**

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuar Profesor

io es

- a) En el direccionamiento inmediato el dato se encuentra en la propia instrucción
- b) En el direccionamiento implícito no se indica la ubicación del operando
- c) El direccionamiento indirecto indica la dirección del operando.
- ✓ d) El direccionamiento indexado es útil para manejo de vectores.

Nombre:
DNI:
Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.
 Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Cuál es el complemento a 2 del número binario 1110 1101 1000?

- a. 0001 0010 0110
- b. 0001 0010 0101
- c. 0001 0010 0111
- d. **0001 0010 1000**

d. 0xfffffffffffffdde1

5. Para comprobar si el contenido del registro RDX es 0 (y posiblemente saltar a continuación usando la instrucción je), el compilador gcc genera:

- a. cmpq %rdx, %rdx
- b. testq %rdx
- c. **testq %rdx, %rdx**
- d. cmpq %rdx

6. Sabiendo que las instrucciones de salto condicional codifican la dirección de salto con direccionamiento relativo a contador de programa (de 8 o 32 bits con signo), indicar cuál es la dirección de salto de la instrucción je en el siguiente desensamblado, donde se ha tachado precisamente dicha dirección.

```
40042f: 74 f4      je xxxxxxxx
400431: 5d          pop %rbp
```

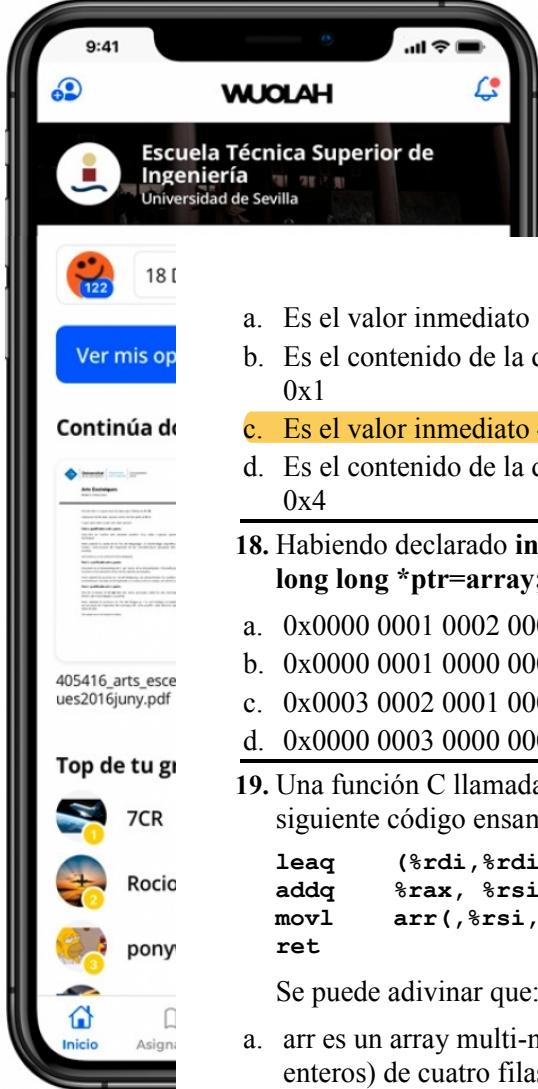
- a. 400431
- b. 400525
- c. **400425**
- d. 40043d

7. Para traducir una asignación condicional (**a = b ? c : d ;**) de lenguaje C a lenguaje ensamblador, gcc puede que utilice...

- a. Un salto incondicional, según la condición expresada en el código C, y otro salto incondicional
- b. Un salto condicional, según la condición opuesta a la del código C, y otro salto condicional



- c. Una instrucción de movimiento condicional, pero sólo si el procesador es Pentium Pro/II o superior
- d. Una instrucción de movimiento incondicional, pero sólo si el S.O. es de 64bits
-
- 8. La instrucción `cmovb %rdx,%rax`**
- copia el byte bajo de rdx en el byte bajo de rax
 - copia en rax el byte de memoria apuntado por la dirección contenida en rdx
 - copia en rax el contenido de rdx si rax es menor que rdx
 - copia en rax el contenido de rdx si CF= 1**
-
- 9. Uno de los puntos clave de la traducción que gcc hace de una construcción switch-case de lenguaje C a lenguaje ensamblador es...**
- el salto condicional hacia atrás
 - el salto relativo a contador de programa
 - el salto directo
 - el salto indirecto**
-
- 10. El procesador utiliza el puntero de pila...**
- En las instrucciones de llamadas y retornos de subrutinas**
 - En todo tipo de instrucciones de saltos, incluyendo llamadas y retornos a subrutinas
 - En todas las instrucciones que tengan al menos dos accesos a memoria
 - En todas las instrucciones
-
- 11. ¿Cuál de las siguientes instrucciones situada al principio de una función se utilizará probablemente para crear espacio en la pila para variables locales sin inicializar?**
- sub \$0x30, %rsp**
 - add \$0x30, %rsp
 - sub \$0x30, %rbp
 - add \$0x30, %rbp
-
- 12. En la convención de llamada SystemV AMD64 seguida por gcc Linux/x86-64...**
- RAX es un registro salva-invocante, por eso en cualquier función hay que salvarlo antes de modificarlo
 - R10 es un registro salva-invocante, por eso si es necesario hay que salvarlo antes de llamar a función**
- caller saved - salva invocante
callee saved - salva invocado*
-
- c. R11 es un registro salva-invocado, por eso en cualquier función hay que salvarlo antes de modificarlo
- d. RBP es un registro salva-invocado, por eso si es necesario hay que salvarlo antes de llamar a función
-
- 13. Un procedimiento llamado por una instrucción call debe guardar y restaurar los registros siguientes siempre que los altere:**
- %rsi, %ordi
 - %rax, %rbx, %rcx, %rdx
 - %rax, %rdx, %rcx
 - %rbx, %rbp**
-
- 14. Dada una función que devuelve la suma de 8 enteros en x86-64, ¿cuál de las siguientes instrucciones suma el 7º argumento?**
- add -0x8(%rsp), %eax
 - add 0x8(%rsp), %eax**
 - add -0x4(%rsp), %eax
 - add 0x4(%rsp), %eax
-
- 15. En el fragmento de programa siguiente:**
- ```
66b: e8 8a ff ff ff callq 5fa <f>
670: 48 83 c4 10 add $0x10,%rsp
```
- ¿Cuál es el valor que introduce en la pila la instrucción callq?
- 0x670**
  - 0xffffffff8a
  - 0x66b
  - 0x5fa
- 
- 16. En el fragmento de programa siguiente:**
- ```
66b: e8 8a ff ff ff  callq 5fa <f>
670: 48 83 c4 10      add $0x10,%rsp
```
- la instrucción callq suma al contador de programa la cantidad:
- 0x76**
 - 0x5fa
 - 0xffffffff8a
 - 0x76
-
- 17. Suponga la siguiente llamada a una función f de 4 argumentos:**
- ```
mov $0x1, %ecx
mov $0x2, %edx
mov $0x3, %esi
mov $0x4, %edi
callq 5fa <f>
```
- El primer parámetro de llamada a la función:



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

- a. Es el valor inmediato 1
  - b. Es el contenido de la dirección de memoria 0x1
  - c. Es el valor inmediato 4
  - d. Es el contenido de la dirección de memoria 0x4
- 

- 18.** Habiendo declarado **int array={0,1,2,3};** y **long long \*ptr=array;** ¿cuánto vale **ptr[1]**?
- a. 0x0000 0001 0002 0003
  - b. 0x0000 0001 0000 0000
  - c. 0x0003 0002 0001 0000
  - d. 0x0000 0003 0000 0002
- 

- 19.** Una función C llamada **get\_el(...)** genera el siguiente código ensamblador.

```
leaq (%rdi,%rdi,4), %rax
addq %rax, %rsi
movl arr(%rsi,4), %eax
ret
```

Se puede adivinar que:

- a. arr es un array multi-nivel (punteros a enteros) de cuatro filas
  - b. arr es un array multi-nivel pero no se pueden adivinar las dimensiones
  - c. arr es un array bidimensional de enteros, no se pueden adivinar dimensiones
  - d. arr es un array bidimensional de enteros, con cinco columnas
- 

- 20.** Las microoperaciones de la fase de captación de una instrucción:

- a. Son comunes para todas las instrucciones
  - b. Dependen del código de operación de la instrucción que se encuentra en el registro de instrucción
  - c. Dependen de los indicadores de estado y del código de operación de la instrucción que se encuentra en el registro de instrucción
  - d. Dependen del valor del contador de programa
- 

- 21.** Para el procesador con unidad de control microprogramada estudiado en clase, Tanenbaum propone codificar los 16 registros y añadir una señal "PERC" para habilitar la carga desde el bus C (recordar que era un diseño típico con 3 buses) y así no perder expresividad/paralelismo. El ahorro de bits en cada microinstrucción debido a esta técnica es de

- a. 40 bits
  - b. 39 bits
  - c. 35 bits
  - d. 29 bits
- 

- 22.** Un procesador está segmentado en k etapas. Cada una de ellas consume un tiempo t. La aceleración ideal (si no hay riesgos) al ejecutar 5 instrucciones respecto a un procesador no segmentado será:

- a. 5k / (4+k)
  - b. (4+k) / 5t
  - c. 4k / (5+k)
  - d. (5+k) / 4t
- 

- 23.** ¿Cuál de las siguientes afirmaciones sobre la E/S programada con consulta de estado es cierta?

- a. Si se emplea E/S programada puede hacerse con consulta de estado o sin consulta de estado
  - b. Un programa que realice salida programada con consulta de estado no ejecutará ninguna instrucción de entrada o carga
  - c. Sólo la E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S
  - d. La escritura de un led requiere consulta de estado
- 

- 24.** En un sistema de interrupciones vectorizado y en daisy-chain, ¿cuál de las siguientes afirmaciones es cierta?

- a. El procesador informa de un ciclo de reconocimiento de interrupción con la señal de reconocimiento de interrupción (INTA) y la identificación de los dispositivos se realiza por consulta de estado
  - b. La gestión de prioridades queda establecida por el orden en que los dispositivos reciben la señal INTA y el dispositivo se identifica por un dato que deposita en el bus
  - c. La gestión de prioridades queda establecida por el orden en que los dispositivos reciben la señal INTA y la identificación de los dispositivos se realiza leyendo sus registros de estado
  - d. El daisy-chain asigna a todos los dispositivos la misma prioridad y la identificación de los dispositivos se realiza leyendo sus registros de estado
-

**25.** ¿Cuál de las siguientes características es menos probable que pueda programarse en un canal DMA?

- a. dos direcciones (origen y destino)
  - b. dos tamaños (copia origen y copia destino)
  - c. cuál de las dos direcciones es de E/S (si alguna lo es) en lugar de Memoria
  - d. si se desea producir una IRQ al terminar
- 

**26.** En un computador con una jerarquía de memoria de dos niveles se observa experimentalmente que el tiempo medio de acceso a la memoria es de 300 ns cuando en realidad el tiempo medio de acceso al primer nivel es de 6 ns. Sabiendo que el tiempo de acceso al segundo nivel es de 3 microsegundos, ¿cuál sería aproximadamente el porcentaje de fallos en los accesos al primer nivel?

- a. 90%
  - b. 1%
  - c. 10%
  - d. 99%
- 

**27.** El orden de magnitud del tiempo de acceso a la memoria DRAM de un computador es de:

- a. Picosegundos
  - b. Nanosegundos
  - c. Microsegundos
  - d. Milisegundos
- 

**28.** Una memoria estática tiene un bus de datos de 32 bits y su bus de direcciones es de 20 bits, ¿cuál es su capacidad?

- a. 4 MBytes
  - b. 1 MByte
  - c. 32 MBytes
  - d. 80 GBytes
- 

**29.** En una cache asociativa por conjuntos de  $2^v$  vías con  $2^b$  líneas (marcos de bloque) de  $2^w$  palabras, el gestor de memoria **no** considera como campo (conjunto de bits contiguos con significado o relevancia) los siguientes bits:

- a. últimos w bits (0...w-1) (los menos significativos)
  - b. bits w...w+c-1 (con c=b-v)
  - c. bits w...w+c-1 (siendo  $2^c=n$  conjuntos)
  - d. bits b...b+c-1 (siendo  $2^c=n$  conjuntos)
- 

**30.** Para obtener una única velocidad comparativa final, el benchmark SPEC CPU combina las ganancias en velocidad de ejecución de una serie de tests, respecto a un ordenador de referencia, usando...

- a. la mediana
  - b. la media aritmética
  - c. la media geométrica
  - d. la moda
-

**Nombre:**
**DNI:**
**Grupo:**

### Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

1. ¿Cuál es el valor mínimo (más negativo) que puede tomar un entero de 32 bits en complemento a dos? (el punto se usa como separador)

- a. -2.147.483.647
- b. **-2.147.483.648**
- c. -4.294.967.295
- d. -4.294.967.296

2. ¿Cómo se representa el valor -1 como entero con signo en 14 bits?

- a. 0xFFFF
- b. **0x3FFF**
- c. las respuestas anteriores no son válidas porque usan hexadecimal; habría que usar binario
- d. no se puede porque 14 no es múltiplo de 4

3. ¿Cuál de las siguientes no es una unidad de la arquitectura Von Neumann?

- a. Unidad central de proceso
- b. Memoria principal
- c. Sistema de entrada/salida
- d. **Núcleo del sistema operativo**

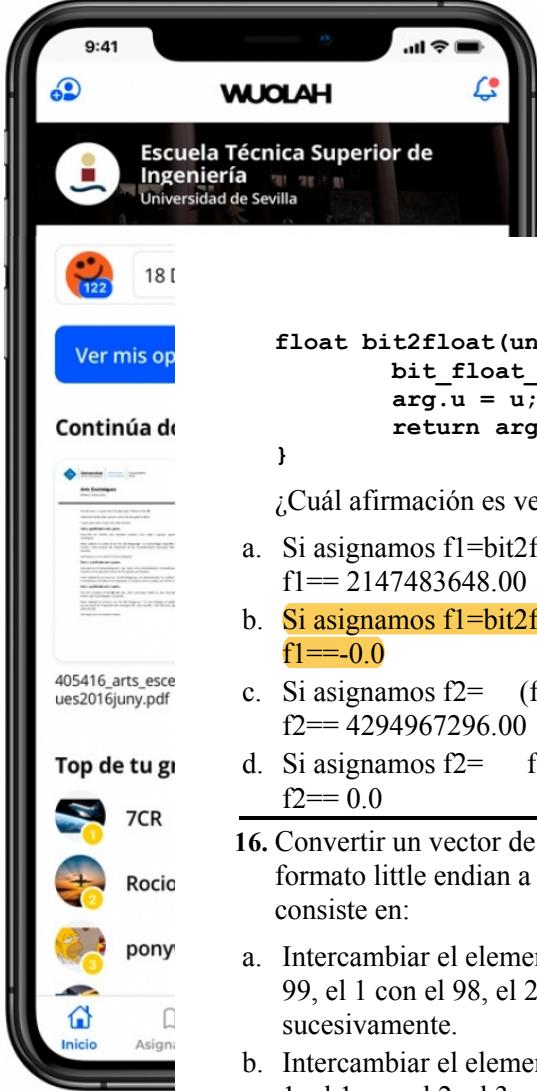
4. ¿Cuál de las siguientes afirmaciones es verdadera?

- a. La arquitectura Von Neumann en la que se basan los computadores tradicionales consiste en tener los datos separados de las instrucciones en memorias distintas.
- b. El registro de estado es un registro transparente al usuario, ya que éste no

puede utilizarlo en las instrucciones máquina.

- c. **El registro de instrucción es un registro transparente al usuario, ya que éste no puede utilizarlo en las instrucciones máquina.**
  - d. La unidad de control necesita como entrada el registro contador de programa, para saber cuál es la instrucción que debe ejecutar a continuación.
- 
5. ¿Qué es el lenguaje máquina?
- a. Conjunto de datos binarios que representan señales eléctricas internas de la unidad de control de un microprocesador.
  - b. Conjunto de sentencias en un lenguaje escrito que se utilizan para generar programas codificados en lenguaje ensamblador.
  - c. Conjunto formado por las siglas asignadas a las instrucciones del repertorio de instrucciones más un conjunto de directivas que facilitan la generación del código binario.
  - d. **Conjunto de instrucciones en formato binario que entiende un determinado procesador.**
- 
6. ¿Cuál de los siguientes elementos no forma parte de la Arquitectura del Repertorio de Instrucciones (ISA)?
- a. Descripción del espacio de direccionamiento de la memoria y de la E/S.

- b. Descripción de los campos de bits en los que están organizadas conceptualmente las microinstrucciones.
- c. Descripción de los registros de datos, registros de estado y control.
- d. Descripción de los tipos de datos sobre los que opera el lenguaje máquina.
- 
7. ¿Cuál de las siguientes definiciones de modos de direccionamiento es **\*incorrecta\***?
- a. Inmediato: el dato está codificado dentro de la propia instrucción, en uno de los campos en los que se divide el formato de instrucción.
- b. Registro: el dato se encuentra en un registro de propósito general.
- c. Directo: la dirección se calcula como la suma de un dato codificado en la propia instrucción y el contenido de un registro de propósito general.
- d. Indirecto: el dato está contenido en una posición de memoria que es apuntada por un registro de propósito general.
- 
8. Respecto a los registros enteros en arquitectura IA32 de 32bits (x86)
- a. Se puede acceder a 8, y en cada uno de esos 8 registros enteros, se puede acceder a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX) ó a los 8 LSBs (p.ej. AL)
- b. Se puede acceder a 8, y en cada uno de esos 8 registros enteros, se puede acceder a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX), a los 8 LSBs (p.ej. AL) o a los bits 8-15 (p.ej. AH)
- c. Se puede acceder a 8 de cada tamaño (32, 16, 8 bits), aunque no todos los registros tienen versión de 8 y 16 bits
- d. No hay distintos tamaños, son sólo registros de 32 bits, como corresponde a dicha arquitectura
- 
9. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):
- a. movw %dx, (%eax)
- b. movb \$0xFF, (%dl)
- c. movswl (%eax), %edx
- d. movzbl %dl, %eax
- 
10. ¿Qué modo de direccionamiento usa el operando fuente en la instrucción mov (%rcx), %al?
- a. Directo a memoria
- b. Indirecto a memoria a través de registro
- c. Registro
- d. Inmediato
- 
11. Si el contenido del registro %rax es 0x10 antes de ejecutar la instrucción shl \$0xc,%rax, ¿cuánto es su contenido tras ejecutarla?
- a. 0x10000
- b. 0x1000
- c. 0x4000
- d. 0x800
- 
12. En el fragmento de código
- ```
804854e: e8 3d 06 00 00 call 8048b90
8048553: 50           pushl %eax
```
- la instrucción call suma al contador de programa la cantidad:
- a. 0x0000063d
- b. 0x08048553
- c. 0x0804854e
- d. 0x50
-
13. ¿Cuál de los siguientes registros x86-64 es distinto del resto en convenio de uso? (salva-invocante/invocado)
- a. RBX
- b. RCX
- c. RSI
- d. R8
-
14. Respecto a requisitos de alineamiento de structs en gcc/IA32 x86 y x86-64, una de las siguientes afirmaciones es ***FALSA***
- a. en x86 Linux alinea double a 4x
- b. en x86 Linux alinea long double a 4x
- c. en x86-64 Linux alinea double a 8x
- d. en x86-64 Linux alinea float a 8x
-
15. Se definen las variables, unión y función C siguientes:
- ```
float f1;
unsigned u1=0x80000000;
float f2;
typedef union {
 float f;
 unsigned u;
} bit_float_t;
```



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**19. ¿Qué es un controlador de E/S?**

- Un circuito electrónico que implementa la memoria del computador.
- Un circuito impreso del tipo DIMM.
- Un circuito electrónico que puede guardar temporalmente datos enviados desde el procesador al periférico o viceversa.
- Un bus que permite interconectar distintos periféricos entre sí.

**20. Respecto a la interfaz de E/S, ¿cuál de las siguientes afirmaciones es \*FALSA\*?**

- Involucra tareas que se pueden implementar parte en hardware y parte en software.
- Permite configurar el funcionamiento del periférico en un momento determinado, y además conocer su estado.
- Puede guardar temporalmente en registros internos tanto datos generados por el periférico para ser enviados al procesador, como datos que son enviados desde el procesador al periférico.
- Una interfaz de entrada recibe los datos desde el procesador y los transforma y envía al periférico en formato digital.

**21. ¿Cuál de las siguientes características corresponde a E/S mapeada en memoria?**

- Determinadas zonas del espacio de direccionamiento del procesador se asignan por convenio a controladores de E/S.
- Un ejemplo de mecanismo de E/S mapeada en memoria es la instrucción IN de los procesadores Intel.
- Una misma dirección se usa alternativamente para E/S y para memoria en distintos momentos de ejecución de un programa.
- Un pin IO/M# del procesador permite distinguir si accedemos a E/S o a memoria.

**22. ¿Cuál de las siguientes afirmaciones es \*FALSA\*?**

- La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa polling para determinar el origen de la IRQ).
- Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó).

- c. Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S.
- d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S.

**23.** La instrucción máquina DI (Disable Interrupts), conocida como CLI (Clear Interrupt Flag) en x86, se utiliza para desactivar:

- a. Todas las interrupciones enmascarables
- b. Las interrupciones de inferior o igual prioridad a una dada
- c. Determinados niveles de interrupción de forma selectiva
- d. Las interrupciones software

**24.** ¿Cuál de los siguientes es un registro de un controlador de DMA?

- a. IR (Instruction Register)
- b. PC (Program Counter)
- c. SP (Stack Pointer)
- d. WC (Word Count)

**25.** El ancho de banda de memoria es:

- a. el número de bits que se pueden transferir entre ésta y la CPU en paralelo en una sola operación de lectura o escritura
- b. el número de bytes que se pueden leer/escribir por unidad de tiempo
- c. el tiempo que se tarda en transferir una palabra entre memoria y CPU
- d. el intervalo de frecuencias de reloj permitidas entre memoria y CPU

**26.** ¿Cuál de las siguientes afirmaciones sobre la memoria DRAM es **\*incorrecta\***?

- a. El principio de funcionamiento de los circuitos electrónicos de la memoria DRAM consiste en cargar o descargar un transistor.
- b. Los bits de memoria se organizan dentro del circuito integrado en forma de matriz de celdas de bit, en la que se pueden diferenciar filas y columnas.
- c. Un transistor en cada celda permite o no permite circular la corriente eléctrica a través de él. Cuando el transistor no deja pasar la corriente, la información queda almacenada durante un tiempo en el condensador. Cuando el transistor deja pasar corriente, el condensador se carga o se descarga.
- d. Cada celda de memoria está compuesta por un transistor y un condensador y almacena un bit de información.

**27.** ¿Cuál de las siguientes afirmaciones sobre memorias es correcta?

- a. La memoria cache se construye con tecnología electrónica de tipo DRAM.
- b. La memoria principal se construye con tecnología electrónica de tipo SRAM.
- c. Los chips de memoria DRAM se conectan entre sí en un circuito impreso constituyendo lo que se denomina DIMM.
- d. Las memorias SRAM no son volátiles; es decir, cuando no están alimentadas eléctricamente siguen guardando toda la información.

**28.** ¿Cuál de los siguientes grupos de señales no se usa en un chip de memoria SRAM?

- a. Selección de filas RAS# y de columnas CAS#.
- b. Datos D<sub>n-1</sub>-D<sub>0</sub>.
- c. Direcciones A<sub>n-1</sub>-A<sub>0</sub>.
- d. Selección de chip CS# y habilitación de escritura WE#.

**29.** ¿Qué es el tiempo de refresco de memoria?

- a. La cantidad de datos transferidos por segundo entre dos niveles de la jerarquía de memoria.
- b. El tiempo que se tarda en recargar los condensadores que almacenan los bits de datos para que no se pierdan.
- c. El tiempo que transcurre entre la solicitud de una operación en un determinado nivel de la jerarquía de memoria (lectura o escritura) y la recepción de todos los datos solicitados.
- d. El tiempo que tiene que transcurrir entre sucesivas solicitudes de acceso a un determinado nivel de la jerarquía de memoria.

**30.** ¿Cuál de las siguientes políticas está **\*menos\*** relacionada con la jerarquía memoria?

- a. Política de escritura: determina cómo se actualiza el nivel de la memoria i+1 cuando se ejecutan instrucciones de almacenamiento en el nivel i.
- b. Política de reemplazo: qué bloque se tiene que sustituir (reemplazar) cuando se trae un bloque desde otro nivel.
- c. Política de planificación: en qué orden se ejecutarán los procesos pendientes.
- d. Política de colocación: dónde se almacena un bloque de datos dentro de la memoria.

# EDEN

El direccionamiento relativo a registro base utiliza...

Usuar Profesor

io es

- a) dos registros.
- b) un registro.
- c) dos desplazamientos contenidos en la propia instrucción.
- ✓ • d) un registro y un desplazamiento.

Puntuación: **1,00**

Una instrucción de salto si menor, para números positivos sin signo, tiene que comprobar el valor de:

Usuar Profesor

io es

- a) los bits de signo y desbordamiento
- b) el bit de signo
- ✗ • c) el bit de cero
- d) el bit de acarreo

Puntuación: **-0,33**

La instrucción setg %al:

U Pro  
su fes  
ari ore  
o s

- a Pone siempre AL a 1.  
)
- ✓ • b Pone AL a 1 en algunos casos.  
)
- c No cambia el contenido de AL  
)
- d Complementa AL si el resultado  
) de la comparación anterior es A >  
B.

Puntuación: **1,00**

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

Usuar Profesor

io es

- a) RET
- b) CALL dir
- c) JNE dir
- d) JMP dir

Puntuación: **1,00**

El lenguaje máquina es...

Usuar Profesor

io es

- a) fácilmente legible por el programador.
- b) portable entre arquitecturas.
- c) difícil de codificar manualmente.
- d) una alternativa razonable al uso del lenguaje ensamblador.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la segmentación de cauce?

Usuar Profesor

io es

- a) primera
- b) segund  
a
- c) tercera
- d) cuarta

Puntuación: **1,00**

[T1.5]

¿En qué generación, dentro de la historia de los computadores digitales, se alcanzaron tiempos de conmutación del orden de nanosegundos?

Usuar Profesor

io es

- a) primera

- b) segunda
- c) tercera
- d) cuarta

Puntuación: **1,00**

En IA32, el registro contador de programa se denomina:

Usuar Profesor

io es

- a) PC
- b) RIP
- c) PC  
R
- d) EIP

Puntuación: **-0,33**

En cdecl/x86, ¿cuál de los siguientes registros tiene que ser guardado por la función llamada si es alterado por ésta?

- a) eax
- b) ecx
- c) ebx
- d) edx

¿Cuál de las siguientes listas está correctamente ordenada temporalmente?

- a) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
- b) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
- c) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
- d) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX

Si el contenido de EBP es 0x13000, ¿a qué dirección se hace referencia con la instrucción INC –0x80(%EBP)?

- ✓ a) 0x13080
- b) 0x13000
- c) 0x80
- d) 0x12F80

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

- ✗ a) 53000h
- b) **4C000h**
- c) 15025h
- d) 5C000h

### 7 [T1.2]

El modo de direccionamiento en el que se especifica un registro y una dirección de memoria cuyo contenido se suma al contenido del registro base para obtener la dirección efectiva, se conoce como:

ón Us Prof  
ún ua eso  
ic río res

- a a base con
  - ) desplazamiento
- b directo o absoluto
  - )
- ✗ c indirecto a registro
  - ) través de memoria
- d ninguno de los
  - ) anteriores

Puntuación: **-0,33**

**8**

[T1.5]

Elección única

Usuar Profesor

io es

- a) primera
- b) segund
- a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

Sobre el ensamblador:

Usuar Profesor

io es

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
- ✓ • d) Todas las respuestas son ciertas.

Suponiendo que todos los registros inicialmente contienen el valor 1 y que el destino es el primer registro, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4
mov r2, #3
add r3, r1, r1
```

sub r1, r3, r2

mul r3, r1, r1

Usuar Profesor

io es

- a) 3
- 6
- b) 6
- ✓ • c) 5
- d) 2
- 0

Puntuación: **1,00**

[T2.2.3]

En los casos concretos indicados para las siguientes instrucciones IA32, ¿cuál no funciona como instrucción de transferencia?

Usuar Profesor

io es

- ✓ • a) cmpl 0x08048040, %eax
- b) movl \$0x15, %eax
- c) pushl %eax
- d) leal variable, %eax

Puntuación: **1,00**

¿Qué medida de prestaciones es la más fiable de todas las posibles?

Usuar Profesor

io es

- a) MIPS
- ✗ b) MIPS equivalentes
- c) ninguna de las otras respuestas es correcta
- d) MFLOPS

Puntuación: **-0,33**

En un microprocesador de 4 bits, una operación en la que el bit 0 de un registro se copia en el acarreo, después el bit 1 se copia en el bit 0, después el bit 2 se copia en el bit 1, y por último el bit 3 se copia en el bit 2, es:

Usuar Profesor

io es

- a) Un desplazamiento lógico a la derecha.
- b) Un desplazamiento aritmético a la derecha.
- c) Una rotación a la derecha.
- d) Una rotación a la derecha a través de acarreo.

Puntuación: **-0,33**

El ancho de palabra de una memoria corresponde a:

Usuar Profesor

io es

- a) La longitud del registro de datos de la memoria.
- b) El número de posiciones que la componen.
- c) El número que identifica únicamente cada posición de la memoria.
- d) La longitud del registro de direcciones de la memoria.

Puntuación: **-0,33**

Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro

Usuar Profesor

io es

- a) De direcciones
- b) Puntero de pila
- c) Contador de programa
- d) De instrucción

Puntuación: **1,00**

El objetivo de un diseño CISC es...

Usuar Profesor

io es

- a) disminuir el tamaño medio de instrucción.
- b) disminuir la frecuencia de reloj.
- c) disminuir el número de instrucciones a ejecutar por un programa.
- d) disminuir el número medio de ciclos por instrucción.

Puntuación: **1,00**

En la captación de la instrucción:

Usuar Profesor

io es

- a) en MBR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- b) en MAR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- c) en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción.
- d) en MBR indicamos la dirección donde está la instrucción y en MAR recogemos la instrucción.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparecen las memorias de semiconductores?

Usuar Profesor

io es

- a) Tercera generación.
- b) Cuarta generación.
- c) Segunda generación.
- d) Quinta generación.

Puntuación: **-0,33**

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

Usuar Profesor

io es

- a) BP

- b) SS \* 10h +  
SP
- X** c) SP
- d) SS \* 10h +  
BP

Puntuación: **-0,33**

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

Usuar Profesor

io es

- ✓**
- a) directo a registro
  - b) a registro inmediato
  - c) a registro literal
  - d) ninguno de los anteriores

Puntuación: **1,00**

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

Usuar Profesor

io es

- ✓**
- a) directo a registro
  - b) a registro inmediato
  - c) a registro literal
  - d) ninguno de los anteriores

Puntuación: **1,00**

Si AX = 0xFA50 y ejecutamos AND \$0xFF, %AX

Usuar Profesor

io es

- X**
- a) El registro AH se pone a FF
  - b) El registro AH se pone a 0
  - c) El registro AL se pone a FF
  - d) El registro AL se pone a 0

Puntuación: **-0,33**

En una máquina con 32 registros direccionables e instrucciones de 16 bits:

Usuar Profesor

io es

**X**

- a) no se pueden codificar a la vez 64 instrucciones de 0 direcciones, 63 instrucciones de dos registros y 16 instrucciones de 1 registro.
- b) no se pueden codificar a la vez 63 instrucciones de dos registros, una instrucción de 0 direcciones y 32 instrucciones de 1 registro.
- c) no se pueden codificar a la vez 64 instrucciones de 1 registro, 32 instrucciones de dos registros y 32 instrucciones de 0 direcciones.
- d) no se pueden codificar a la vez 62 instrucciones de dos registros, 32 instrucciones de 1 registro y 64 instrucciones de 0 direcciones.

Puntuación: **-0,33**

**7**

¿Qué tipos de instrucciones se emplean más en una arquitectura de registros de propósito general?

Elecció  
n única

Usua  
rio Profes  
ores

**X**

- a) De desplazamiento y rotación.
- b) Aritmético-lógicas.
- c) De transferencia de datos.
- d) De transferencia de control.

Puntuación: **-0,33**

8

Elecció n únic a Si d es un desplazamiento, r un registro índice e i una constante apropiada, el modo de direccionamiento indexado con postautodecremento realiza...

Usua rio  
n  
únic a  
Profes ores



- a) dirección efectiva =  $r + d$ ;  $r = r - i$
- b) dirección efectiva =  $r - i$ ;  $r = r - d$
- c)  $r = r - i$ ; dirección efectiva =  $r - d$
- d)  $r = r + i$ ; dirección efectiva =  $r - d$

Puntuación: 1,00

En el direccionamiento inmediato:

Usuar Profesor

io es



- a) el tamaño (rango de valores) de la constante está limitado.
- b) el operando es una constante de tamaño arbitrario contenida en la misma instrucción.
- c) el código de operación indica el operando afectado.
- d) una vez captada y decodificada la instrucción, se accede a memoria para obtener el valor del operando.

Puntuación: 1,00

En el RISC-I, una ventana de registros contiene:

Usuar Profesor

io es



- a) registros para recibir parámetros del procedimiento llamador.
- b) registros para almacenar variables locales.
- c) registros para enviar parámetros a procedimientos.
- d) todas las respuestas son ciertas.

Puntuación: -0,33

Si queremos almacenar la palabra de 16 bits 0x9660 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x1000 como:

Usuar Profesor

io es

- a) M[0x1000]=0x06 y  
M[0x1001]=0x69
- b) M[0x1000]=0x69 y  
M[0x1001]=0x06
- ✓ • c) M[0x1000]=0x60 y  
M[0x1001]=0x96
- d) M[0x1000]=0x96 y  
M[0x1001]=0x60

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?

Usuar Profesor

io es

- a) primera
- b) segund  
a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

GCC/Linux IA32 resuelve el ajuste de marco de pila al comenzar un procedimiento mediante las instrucciones:

Usuar Profesor

io es

- a) pushl %ebp movl %esp,  
%ebp
- b) movl %esp, %ebp popl %esp
- ✗ c) pushl %esp movl %ebp,  
%esp
- d) movl %ebp, %esp popl %ebp

Puntuación: **-0,33**

¿En qué orden debería ejecutarse en una máquina de tipo pila la operación aritmética ( $a+b/c-d$ )?

Usuar Profesor

io es

a)  $a + b / c -$

d

b)  $a b + c d -$

/



- c)  $a b c / + d$

-

d)  $a b + / c d$

-

Puntuación: **1,00**

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

Usuar Profesor

io es

a) registro-registro

b) registro-memoria



- c) memoria-memoria

d) ninguna de las anteriores es cierta

Puntuación: **1,00**

¿Cuál de las siguientes parejas de microprocesadores representa mejor los conceptos RISC?

Usuar Profesor

io es



- a) MIPS, SPARC

b) PA-RISC,  
PowerPC

c) Itanium, Alpha

d) Pentium, Athlon

Puntuación: **1,00**

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

Usuar Profesor

io es

- a) SS \* 10h +  
    BP
- b) BP
- c) SS \* 10h +  
    SP
- X d) SP

Puntuación: **-0,33**

**9** [T1.5]

Elecció  
n única

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?

Usuar Profesor

io es

- a) primera
- b) segunda
- ✓ c) tercera
- d) cuarta

Puntuación: **1,00**

**2**

[T1.1]

Elección  
única

Usu Profe  
ario sores

- ✓ • a de transferencia de datos con  
    ) memoria
- b aritmético-lógicas  
    )
- c de transferencia de datos  
    ) entre registros

d de desplazamiento y rotación  
)

Puntuación: **1,00**

Una máquina superescalar es aquella que:

Usuar Profesor

io es

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- ✓ • c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
- d) ninguna respuesta de las anteriores es correcta.

Puntuación: **1,00**

¿En qué generación, dentro de la historia de los computadores digitales, aparece la segmentación de cauce?

Usuar Profesor

io es

- a) primera
- b) segund
- a
- ✓ • c) tercera
- d) cuarta

Puntuación: **1,00**

Si el registro EAX contiene X, La secuencia de instrucciones siguiente:

cmpl \$6, %eax

jae Destino

salta a la etiqueta Destino sólo si:

Usuar Profesor

io es

- ✓ • a)  $X < 0 \text{ || } X \geq 6$
- b)  $X \leq 6$
- c)  $X \geq 0 \text{ && } X \leq 6$
- d)  $X > 6$

Puntuación: **1,00**

El lenguaje máquina es...

Usuar Profesor

io es

- ✓ • a) difícil de codificar manualmente.  
b) portable entre arquitecturas.  
c) fácilmente legible por el programador.  
d) una alternativa razonable al uso del lenguaje ensamblador.

Puntuación: **1,00**

Si usamos una estructura de bus con DMA:

Usuar Profesor

io es

- a) al bus del sistema sólo se conecta la CPU y la MP, ya que el DMA se conecta directamente a MP para realizar las transferencias de datos.
- ✗ b) podemos prescindir de controladores de E/S ya que el controlador de DMA se ocupa de controlar las transferencias hacia/desde los periféricos.
- c) la velocidad de este controlador establece la velocidad del bus del sistema.
- d) la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.

Puntuación: **-0,33**

Sobre el direccionamiento relativo al contador de programa:

Usuar Profesor

io es

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está ejecutando.
- ✓ • d) Todas las respuestas son ciertas.

Puntuación: **1,00**

Si %eax contiene x, ¿cuál de las siguientes instrucciones calcula x\*5?

Usuar Profesor

io es

- a) leal 3(%eax,2),%edx
- b) leal 3(%eax,%eax,2),%edx
- c) leal 4(%eax,%eax),%edx
- d) leal (%eax,%eax,4),%edx

Puntuación: **1,00**

Si queremos almacenar la palabra de 16 bits 8965h en memoria según "big-endian", quedará almacenada a partir de la posición 1000h como:

Usuar Profesor

io es

- a) en el byte 1000h se guarda A6h y en el 1001h 91h
- b) en el byte 1000h se guarda 91h y en el 1001h A6h
- c) en el byte 1000h se guarda 89h y en el 1001h 65h
- d) en el byte 1000h se guarda 65h y en el 1001h 89h

Puntuación: **1,00**

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

Usuar Profesor

io es

- a) 15025h
- b) 4C000 h
- c) 5C000 h
- d) 53000h

Puntuación: **1,00**

Sobre el ensamblador:

Usuar Profesor

io es

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
- ✓ • d) Todas las respuestas son ciertas.

Puntuación: **1,00**

Un procesador emplea codificación en bloque del código de operación. Existen 123 instrucciones que tienen una longitud de 16 bits. ¿A cuántas direcciones de memoria pueden acceder como máximo si todas emplean una estructura “código de operación + dirección de memoria”?

Usuar Profesor

io es

- a) 128.
- ✗ b) 256.
- c) 1024
- .
- d) 512.

Puntuación: **-0,33**

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

Usuar Profesor

io es

- a) registro-registro
- b) registro-memoria
- ✓ • c) memoria-memoria
- d) ninguna de las anteriores es cierta

Puntuación: **1,00**

Para direccionar una memoria de 1 G palabras de 32 bits cada una, que se direcciona byte a byte, se necesitarán:

Usuar Profesor

io es

**X**

- a) 33 bits
- b) 21 bits
- c) 32 bits
- d) 31 bits

Puntuación: **-0,33**

[T1.5]

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria virtual?

Usuar Profesor

io es

**✓**

- a) primera
- b) segund a
- c) tercera
- d) cuarta

Puntuación: **1,00**

En el RISC-I, una ventana de registros contiene, entre otros registros,...

Usuar Profesor

io es

**✓**

- a) registros para recibir parámetros del procedimiento llamador
- b) registros para almacenar matrices de enteros
- c) registros para enviar parámetros a otros procesos
- d) todas las respuestas son falsas

Puntuación: **1,00**

En el arbitraje de un bus...

Usuar Profesor

io es

- a) los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
- b) si hay un único dispositivo pasivo, siempre funciona como esclavo
- c) si hay varios dispositivos activos, siempre funcionan como maestros
- X d) todas las respuestas anteriores son ciertas

Puntuación: **-0,33**

¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?

Usuar Profesor

io es

- a) posición 0: FF pos.1:FF pos.2: FF pos.3:  
80
- ✓ • b) 0:80 1:FF 2:FF 3:FF
- c) 0:00 1:01 2: 00 3:80
- d) Ninguna de las anteriores

Puntuación: **1,00**

Si A=FF0Fh y B=0004h, el resultado de desplazar A a la derecha aritméticamente B veces es:

Usuar Profesor

io es

- a FFF0h  
)
- X b 0FF0h  
)
- c F0FFh  
)
- d F0F0h  
)

Puntuación: **-0,33**

Suponiendo que todos los registros inicialmente contienen el valor 0, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4
mov r2, #3
add r3, r1, r1
sub r1, r3, r2
mul r3, r1, r1
```

Usuar Profesor

io es

- a) 4
- b) 5
- c) 2
- 5
- d) 0

Puntuación: **1,00**

Un modo de vídeo de 1680 x 1050 píxeles y 4 canales de color por píxel (R, G, B, alpha) y 256 niveles de intensidad por canal (0 a 255), ocupa aproximadamente una memoria de:

Usuar Profesor

io es

- a) 1,7 MB
- b) 7 MB
- c) 54 MB
- d) 1,7 GB

Puntuación: **1,00**

Suponiendo que todos los registros inicialmente contienen el valor 0, ¿cuál es el valor de r1 tras la ejecución de la siguiente secuencia de instrucciones?

```
mov r1, #4
mov r2, #3
add r3, r1, r1
sub r1, r3, r2
mul r3, r1, r1
```

Usuar Profesor

io es

- a) 4
- b) 5
- c) 25
- d) 0

Puntuación: **1,00**

Una máquina superescalar es aquella que:

Usuar Profesor

io es

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
- X d) ninguna respuesta de las anteriores es correcta.

Puntuación: **-0,33**

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuar Profesor

io es

- a) En el direccionamiento inmediato el dato se encuentra en la propia instrucción
- b) En el direccionamiento implícito no se indica la ubicación del operando
- ✓ • c) El direccionamiento indirecto indica la dirección del operando.
- d) El direccionamiento indexado es útil para manejo de vectores.

Las arquitecturas de acumulador se caracterizan por:

Usuar Profesor

io es

- X
- a) tráfico reducido entre procesador y memoria.
  - b) instrucciones muy largas.
  - c) diseño simple del procesador.
  - d) facilidad de generación de código eficiente para los compiladores.

Puntuación: **-0,33**

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

Usuar Profesor

io es

- ✓
- a) SS \* 10h +  
SP
  - b) BP

c) SS \* 10h +

BP

d) SP

Puntuación: **1,00**

¿Qué modelo de programa se ejecuta en las arquitecturas de Von Neumann?

Usuar Profesor

io es

a) nanoprogramado

b) cableado

c) micropogramado

✓ • d) almacenado

Puntuación: **1,00**

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| b | c | c | b | c |

**1.** El programador de lenguaje ensamblador necesita conocer:

- a. la microarquitectura del procesador
- b. la arquitectura del procesador
- c. el diseño RTL del procesador
- d. la organización del procesador (estructura y funcionamiento)

Ver Tema2.1, tr.14

**2.** La diferencia entre el flag de acarreo y de overflow es que...

- a. uno se activa para números con signo y otro para números sin signo
- b. el flag de acarreo indica que ha habido acarreo en una operación con números enteros, el de overflow indica que ha habido desbordamiento en una operación con números en punto flotante
- c. ambos se recalculan tras cada operación aritmético-lógica con enteros, correspondiendo al programador consultar uno u otro según piense que sus datos son con signo o sin signo      **Ver manual Intel ADC, o Vol1. Secc. 3.4.3 EFLAGS**
- d. el de acarreo indica que el resultado es demasiado grande (si se trata de punto flotante) o positivo (si se trata de enteros) para poder almacenarse, el de overflow indica que es demasiado pequeño (punto flotante) o negativo (enteros)

**3.** Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos, y un computador Y tarda 14 segundos en ejecutar el mismo programa, ¿cuántas veces es más rápido el computador Y que el X?

- a. 12
- b. 0.538
- c. 1.857
- d. 1.538

$26/14 = \text{casi } 2, 13/7 = 1.\text{algo}, \text{resto } 6, 8*7=56$

**4.** El direccionamiento directo a memoria utiliza

- a. dos desplazamientos contenidos en la propia instrucción
- b. un desplazamiento      **Ver Tema1, tr.31-33, o Tema2.1, tr.40, o Tema2.2, tr.4-5**
- c. un registro y un desplazamiento contenidos en la propia instrucción
- d. un registro

**5.** Si queremos almacenar la palabra de 64bits 0x00000001f ffffffe0 en una memoria de bytes según la convención “little-endian” a partir de la posición 0x0804913c, quedará almacenado

- a. 0x00 en 0x0804913c y 0xe0 en 0x08049143
- b. 0x1f en 0x0804913c y 0xe0 en 0x08049140
- c. 0xe0 en 0x0804913c y 0x1f en 0x08049140
- d. Todas las respuestas anteriores son incorrectas

orden de LSB a MSB: e0, ff, ff, ff, 1f, 00, 00, 00

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a, b, c o d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| b | b | b | d | c |

**1.** ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):

- a. movw %dx, (%eax)
  - b. pushb \$0xFF
  - c. movswl (%eax), %edx
  - d. movzbl %dl, %eax
- sería pushl, no existe. Ejercicio 3.2, #5

**2.** ¿Cuál de las siguientes secuencias de instrucciones multiplica el (contenido del) registro EAX por 18?

- a. imull \$0x18, %eax
  - b. leal (%eax,%eax,8), %eax  
leal (%eax,%eax), %eax
  - c. shll \$18, %eax
  - d. sarl \$1, %eax  
imul \$9, %eax
- (eax + 8 eax) x 2 = 18 eax, Tema2.2, tr.6

**3.** La instrucción **xor \$3, %eax** tiene como resultado:

- a. Poner a 0 los últimos 3 bits del registro EAX
- b. Cambiar 0↔1 (complemento a 1 de) los últimos 2 bits del registro EAX
- c. Poner a 1 el último bit del registro EAX
- d. Ninguno de los anteriores resultados

**4.** La instrucción IA32 **test** sirve para... d), Tema2.3 tr.22

- a. Testear el código de condición indicado, y poner un byte a 1 si se cumple
- b. Mover el operando fuente al destino, pero sólo si se cumple la condición indicada
- c. Realizar la operación resta (a-b) pero no guardar el resultado, sino simplemente ajustar los flags
- d. Realizar la operación and lógico bit-a-bit (a&b) pero no guardar el resultado, sino simplemente ajustar los flags

**5.** De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...

- a. El bucle for
  - b. El bucle while
  - c. El bucle do-while
  - d. La selección switch-case
- Tema2.3 tr.40,46

**Nombre:****DNI:****Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | b | d | b | a |

1. GCC/Linux IA32 resuelve el ajuste de marco de pila mediante las instrucciones:

- a. `movl %esp, %ebp;`      `popl %esp`
- b. `movl %ebp, %esp;`      `popl %ebp`
- c. `pushl %esp;`              `movl %ebp, %esp`
- d. `pushl %ebp;`              `movl %esp, %ebp`

[Ver Tema2.3 tr.26-28, por ejemplo](#)

2. La instrucción `leave` equivale a:

- a. `movl %esp, %ebp;`      `popl %esp`
- b. `movl %ebp, %esp;`      `popl %ebp`
- c. `pushl %esp;`              `movl %ebp, %esp`
- d. `pushl %ebp;`              `movl %esp, %ebp`

[Ver Tema2.3 tr.32, por ejemplo](#)

3. Alguna de estas opciones contiene algún elemento que no corresponde a los contenidos de un marco de pila GCC/Linux IA32.

- a. Argumentos de llamada a la función y Dirección de retorno
- b. Antiguo marco de pila y Registros salva-invocante
- c. Variables locales y Registros salva-invocado
- d. Variables globales y Valor de retorno de la función

[Ver Tema2.3 tr.24 o 49, por ejemplo](#)

4. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

- a. EAX
- b. EBX
- c. ECX
- d. EDX

[Ver Tema2.3 tr.36](#)

5. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

- a. EAX
- b. EBX
- c. ESI
- d. EDI

[Ver Tema2.3 tr.36](#)

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a, b, c o d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| b | c | d | b | d |

**1.** La arquitectura x86-64 tiene:

- a. 8 registros de propósito general (RPG) de 64 bits (%rax, %rbx, ... %rsp, %rbp)
- b. 16 registros RPG de 64 bits
- c. 32 registros RPG de 64 bits
- d. 64 registros RPG de 64 bits

[Ver Tema2.4 tr.4](#)

**2.** La convención de llamada x86-64 Linux/GCC considera, respecto a convenios de uso de registros:

- a. 3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales
- b. 8 registros salva-invocante, 6 registros salva-invocado, y 2 especiales
- c. Algunos registros salva-invocante, otros salva-invocado, uno especial
- d. Algunos registros para pasar argumentos, otros salva-invocante, otros salva-invocado, dos especiales

[Ver Tema2.4 tr.5](#)

**3.** ¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso?

- a. RAX, RBX, RCX, RDX
- b. RBX, RSI, RDI
- c. RSP, RBP
- d. CL, DX, R8d, R9

[Ver Tema2.4 tr.5, Práctica 1 pags.12-13 y Figura 3.35 libro](#)

**4.** La zona roja en x86-64 Linux es...

- a. una zona de pila en donde no deben escribir las funciones invocadas (porque si se produjera una interrupción, el manejador de interrupción sobreescribiría los valores escritos en pila)
- b. una zona bajo (RSP) (adonde apunta RSP) que una función puede usar sin reservarla, pero sólo si no llama a ninguna otra función mientras la usa
- c. una zona bajo (RBP) (adonde apunta RBP) que pueden usar sólo las funciones que no necesiten que RBP baje más. Por ejemplo, si se necesita almacenar variables locales o salvar registros salva-invocado, no se puede usar la zona roja
- d. una zona bajo (RSP) (adonde apunta RSP) que pueden usar sólo las funciones que no necesiten que RSP baje más. Por ejemplo, si se necesita calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función, no se puede usar la zona roja

[Ver Tema2.4 tr.8](#)

**5.** El marco de pila en x86-64 Linux...

- a. no existe, porque RBP no es registro especial en x86-64
- b. sólo se crea para funciones que invocan anidadamente a otra función (procedimientos padre, no hojas)
- c. se crea para funciones en las que GCC no puede evitar que RBP baje más, como por ejemplo: que haya demasiadas variables locales (y no quepan en registros), o que haya que salvar algún registro salva-invocado
- d. se crea para funciones en las que GCC no puede evitar que RSP baje más, como por ejemplo: que haya que calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función

[Tema2.4 tr.9,10,13 y p.318-319 libro](#)

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | a | c | a | b |

**1.** La diferencia entre “unidad de control” y “unidad de procesamiento” consiste en que:

- a. La primera está incluida en la segunda
- b. La segunda está incluida en la primera
- c. Son unidades distintas, aunque relacionadas
- d. Son distintos nombres para la misma unidad

ver p.ej. Tema3, tr.5

**2.** La diferencia entre “unidad aritmético-lógica” y “unidad de procesamiento” consiste en que:

- a. La primera está incluida en la segunda
- b. La segunda está incluida en la primera
- c. Son unidades distintas, aunque relacionadas
- d. Son distintos nombres para la misma unidad

ver p.ej. Tema3, tr.6

**3.** Alguna de las siguientes **\*no\*** es una operación básica de la unidad de control:

- a. Transferir un registro a otro
- b. (Leer o escribir) un registro (de / a) memoria
- c. (Guardar o recuperar) un registro (en / de) la pila
- d. Realizar operación ALU y guardar resultado en registro

ver Tema3, tr.12

**4.** Es típico que los registros del banco de registros tengan líneas de control como:

- a. “Load” y “Enable”, en unidades de control con un bus
- b. “LoadB”, “EnableB”, “LoadM” y “EnableM”, en unidades con un bus, según se lea/escriba al bus interno o a memoria
- c. “LoadB”, “EnableB”, “LoadM” y “EnableM”, en unidades con buses múltiples, según se lea/escriba al bus interno o a memoria
- d. “Load[A,B,C]” y “Enable[A,B,C]”, por ejemplo (6 en total), en unidades con 3 buses A,B,C, según de qué bus se lea/escriba

ver Tema3, tr.13, 19, 21

**5.** Las líneas de control del registro MDR/MBR en los diseños típicos de unidad de control con uno o con tres buses, son:

- a. 2 “Load” y 2 “Enable” en ambos diseños
- b. 2 “Load” y 2 “Enable” con 1 bus, 2 “Load” y 3 “Enable” con 3 buses
- c. 2 “Load” y 3 “Enable” con 1 bus, 2 “Load” y 2 “Enable” con 3 buses
- d. Ninguna de las anteriores respuestas es correcta

ver Tema3, tr.13, 19, 21

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | d | d | d | d |

**1.** Alguna de las siguientes señales **\*no\*** sirve de entrada a la unidad de control. ¿Cuál?

- a. Señal de reloj (CLK)
- b. Estado de la unidad de proceso (flags Z,S,C,O...)
- c. Instrucción actual (bits del registro IR)
- d. Contador de programa (bits del registro PC)

ver Tema3, tr.25

**2.** Alguna de las siguientes señales **\*no\*** es salida de la unidad de control. ¿Cuál?

- a. Señales de carga, habilitación y/o desplazamiento de registros (Load, Enable, ShiftL, ShiftR)
- b. Códigos de selección en multiplexores, decodificadores, ALU, etc (00,01,10,11...)
- c. Señales de lectura y escritura en memoria (RD, WR)
- d. Dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)      ver Tema3, tr.25

**3.** Respecto a los términos microinstrucción y microcódigo:

- a. Son equivalentes, llamamos microcódigo o microinstrucción a una palabra de la memoria de control
- b. Una microinstrucción está programada en microcódigo, que es un lenguaje para programar señales de control
- c. Un microcódigo controla una serie de señales de control relacionadas (por ejemplo, el código 000 para que la ALU realice la suma), y varios microcódigos juntos forman una microinstrucción
- d. Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoria

**4.** Respecto a los términos micropograma y microcódigo:

- a. Son equivalentes, igual que decimos “un programa C” o “un código (fuente) C”
- b. Microcódigo es el lenguaje, y micropograma es un programa escrito en dicho lenguaje, igual que decimos que “un programa C” está escrito en (o sencillamente, es...) “código C”
- c. Un micropograma está formado por varios microcódigos
- d. Un microcódigo está formado por varios micropogramas      ver Tema3, tr.42

**5.** Parecidos y diferencias entre secuenciamiento explícito e implícito en una UC microprogramada. Alguna de las siguientes afirmaciones **\*no\*** es correcta.

- a. En el implícito hay un microcontador de programa, en el explícito no
- b. En el explícito hay un campo de dirección de la siguiente microinstrucción (o incluso dos, para microbifurcaciones), en el implícito no hay tal campo (o sólo uno, para microbifurcaciones, por lo que se suele solaparlo con otros campos)
- c. En el implícito se necesita una ROM o PLA de traducción codop→dirección 1ª microinstrucción, en el explícito no
- d. Ambos secuenciamientos necesitan un campo de 1 bit “TS” (tipo de salto) con el cual se marca la 1ª microinstrucción de una instrucción máquina, y otro campo de 1bit “TS2” (si la UC hace microbifurcaciones) con el que se marcan justamente las microinstrucciones de salto (microbifurcaciones)      ver Tema3, tr.53-58, y 82,84,86,89

ej. Tanenbaum no tiene TS1,TS2. Además, sería última uinstr, no 1ª

Nombre:

DNI:

Grupo:

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | b | d | a | d |

**1.** La ganancia en velocidad de un cauce de K etapas de igual duración ejecutando un programa de N instrucciones es:

- a.  $S = KN/(K-N+1)$
- b.  $S = NKT/(N-K+1)T$
- c.  $S = KN/(K+N-1)$
- d.  $S = NT/(N+K-1)T$

**Ver Tema4, p.18**

**2.** Suponer que el tiempo (T) de ejecución de una instrucción en un procesador pudiera dividirse en 4 etapas, pero no de igual duración, sino p.ej.: T/3, T/6, T/3, T/6. ¿Cuál sería aproximadamente la ganancia (S) del correspondiente cauce segmentado?

- a. No se puede hacer un cauce segmentado con etapas de distinta duración
- b. Depende de la etapa más lenta, y en este caso sería 3
- c. Depende de la etapa más rápida (en este caso T/3)
- d. Ninguna de las respuestas anteriores

**Ver Tema4, pp.18-19**

**3.** Hay riesgos que afectan a las prestaciones de un cauce segmentado. Alguna de las siguientes respuestas **\*no\*** es uno de ellos:

- a. Conflicto por recursos (que 2 instrucciones o 2 etapas necesiten el mismo recurso al mismo tiempo)
- b. Riesgos de datos (dependencias RAW, etc)
- c. Riesgos de control (huecos de retardo de salto, etc)
- d. Riesgos de direcciones (uso de la misma dirección en 2 instrucciones)

**Ver Tema4, pp.24,28,31**

**4.** Una cola de precaptación sirve para:

- a. Reducir el efecto de los fallos de cache
- b. Disminuir el periodo de ciclo del cauce
- c. Aumentar el número de etapas del cauce
- d. Resolver ciertos problemas de dependencia de datos

**Ver Tema4, p.27**

**5.** Los problemas de dependencias de datos que hemos estudiado... ¿los puede resolver el propio procesador (por diseño de su UC), o el programador de lenguaje máquina, o el compilador que se use?

- a. Sólo los puede resolver el procesador, si implementa adelanto de registros (*register forwarding, RF*).
- b. Sólo los puede resolver el programador (o compilador), insertando operaciones NOP.
- c. Depende de la CPU: si implementa RF no hace falta hacer nada más, aunque tampoco haría daño añadir los NOP necesarios (si existe la operación en el repertorio), y en caso de que no tenga RF sería responsabilidad del programador o compilador añadir los NOP necesarios
- d. Ninguna de las respuestas anteriores

**Ver Tema4, p.30**

Nombre:

DNI:

Grupo:

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a, b, c o d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | c | b | b | d |

**1.** Respeto a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:

- a. La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo ver Tema5 tr.5-6
- b. Cada puerto o interfaz es una línea de comunicación con el procesador. El conjunto de ellos forma el controlador.
- c. El puerto, o interfaz, contiene los controladores necesarios para comunicar el dispositivo con el procesador
- d. El interfaz contiene las controladoras necesarias para conectar los puertos con el procesador

**2.** Respeto a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida:

- a. Todas ellas impiden que haya cortocircuito en el bus de datos
- b. Todas ellas impiden que haya cortocircuito en el bus de direcciones
- c. La selección lineal permitiría escribir un mismo dato a varios puertos E/S ver Tema5 tr.7-9
- d. Usando decodificación centralizada es más fácil realizar expansiones al sistema

**3.** Respeto a las operaciones de lectura y escritura, con temporización síncrona o asíncrona: ver Tema5 tr.11-16

- a. En lectura síncrona, el procesador pone dirección y datos en los buses correspondientes, y espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado datos no (ACK tampoco)
- b. En escritura síncrona, el procesador pone dirección y datos en los buses correspondientes (y también los retira de los buses) en instantes de tiempo determinados por el reloj del bus ok
- c. En lectura asíncrona, el procesador pone los datos en el bus de datos, y espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado datos no
- d. En escritura asíncrona, el procesador pone la dirección en el bus de direcciones (y también la retira) en instantes de tiempo determinados por el reloj del bus reloj bus no

**4.** Respeto a si un computador dispone de E/S independiente (separada) y/o usa E/S mapeada a memoria: Tema5 tr.21-28

- a. Si el encapsulado (chip) del procesador tiene patilla (pin) IO/M# (o patillas equivalentes), eso evidencia que el computador usa E/S mapeada a memoria dispone de E/S separada, otra cosa es que se use o no
- b. Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada
- c. Si el encapsulado del procesador no dispone de patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada E/S mapeada siempre disponible, otra cosa es si se usa o no
- d. Si el repertorio del procesador tiene instrucciones del tipo LOAD y STORE, el computador sólo dispone de E/S mapeada a memoria puede tener además IN y OUT para E/S separada

**5.** Alguna de las siguientes \*no\* es una ventaja de la E/S independiente (separada, aislada)

- a. Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
- b. Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- c. Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- d. Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño) ver Tema5, tr.23,25

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | d | a | b |

1. Respecto a parecidos y diferencias entre los métodos de E/S: Alguna de las siguientes afirmaciones es incorrecta:
  - a. La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa *polling* para determinar el origen de la IRQ)
  - b. Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó)
  - c. Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S    **E/S IRQ de fuente conocida**
  - d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
2. Respecto a los tipos de interrupción que suelen contemplar las CPUs comerciales actuales: Alguno de los siguientes no lo es:
  - a. Internas (excepciones o *traps*): generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, fallo de página, etc)
  - b. Externas (IRQs hardware): generadas por un dispositivo externo a la CPU, activan la línea #INTR (o equivalente)
  - c. Software: generadas al ejecutar la instrucción INT (o equivalente)
  - d. Firmware (*faults*): generadas por el microcódigo de la CPU (*segm. fault, page fault...*)      **ver Tema5 tr.50-51**
3. Alguna de las siguientes técnicas **\*no\*** es de utilidad para determinar la causa de una interrupción      **ver Tema5 tr.56-58**
  - a. Consulta de estado, o *polling*
  - b. Interrupciones vectorizadas
  - c. Múltiples líneas de interrupción INT1#, INT2#...
  - d. *Daisy-chain* (encadenamiento) de ACK#      **útil para prioridad entre IRQs simultáneas, ver Tema5 tr.65-67**
4. En un sistema computador con varios dispositivos que hagan IRQ aparecen algunos problemas que se suelen resolver mediante mecanismos de prioridad. Alguno de los siguientes problemas o técnicas **\*no\*** está relacionado con este escenario:
  - a. Redisparo de interrupciones      **ver Tema5 tr.59**
  - b. Interrupciones simultáneas
  - c. Interrupciones anidadas
  - d. Inhibición de interrupciones
5. Respecto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR):
  - a. No es necesario salvar ninguno más, el contador de programa y los flags de estado ya los salva la propia CPU como parte del mecanismo de interrupción
  - b. Se deben guardar los registros que se modifiquen en la propia ISR. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar      **ver Tema5, tr.46**
  - c. Se deben guardar los registros salva-invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA-32), los registros salva-invocante ya los guarda el programa interrumpido
  - d. Se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción

Nombre:

DNI:

Grupo:

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | c | a | b | a |

**1.** La localidad de las referencias a memoria se manifiesta de diversas formas, pero alguna de las siguientes **\*no\*** es una de ellas:

- a. Localidad espacial
- b. Localidad temporal
- c. Localidad secuencial
- d. Localidad paralela

ver Tema6, tr.15

**2.** Sólo una de las siguientes afirmaciones es correcta. ¿Cuál? (La dirección referenciada en el instante de tiempo  $t$  se indica como  $d(t)$ , y se supone que  $k, n$ , son pequeños, con  $n$  positivo)

- a. La localidad espacial se manifiesta cuando  $d(t + n) = d(t)$
- b. La localidad temporal se manifiesta cuando  $d(t + n) = d(t) + k$
- c. La localidad secuencial se manifiesta cuando  $d(t + 1) = d(t) + 1$
- d. La localidad paralela se manifiesta cuando  $d(t + n) = d(t + k)$

tr.16

**3.** Sólo una de las siguientes afirmaciones sobre memorias ROM es correcta. ¿Cuál?

- a. Para fabricar una ROM se deben conocer los datos que se desea que almacene tr.32-34
- b. Una PROM (*Programmable ROM*) se puede grabar usando un dispositivo programador que selectivamente funde contactos aplicándoles altas temperaturas mediante diminutas cabezas soldadoras ("equipo de puntas")
- c. Una EPROM (*Electrically Progr. ROM*) se puede grabar eléctricamente, sin fundir contactos, pero no se puede borrar
- d. Una EEPROM (*Erasable EEPROM*) se puede grabar (eléctricamente), y borrar (usando rayos ultravioleta)

**4.** Sobre memorias RAM estáticas y dinámicas... ¿cuál afirmación es incorrecta?

- a. Un bit de SRAM (6 transistores) es más costoso que un bit de DRAM (1 transistor + 1 condensador) tr.39-43
- b. También en consumo eléctrico, un bit SRAM es más costoso que uno DRAM, porque hay que refrescarlo frecuentemente (gastando electricidad para ello) al revés, tr.39,46
- c. Desde el punto de vista electrónico, en un bit SRAM la operación difícil es la escritura (no la lectura), porque pueden tener que competir los inversores que almacenan el bit con los *drivers* que fijan el nuevo valor. Para facilitar la escritura, se ataca al bit por ambos inversores ( $Q$  y  $\#Q$ ) y los *drivers* son más anchos que los inversores. tr.40-41
- d. Desde el punto de vista electrónico, en un bit DRAM la operación difícil es la lectura, porque es destructiva: la carga almacenada se dispersa por la línea de lectura de bit. Para facilitar la lectura se usan amplificadores especiales (*sense amplifiers*) y se vuelve a escribir el bit recién destruido tr.43-46

**5.** Respecto a la temporización y cronogramas de funcionamiento de memorias DRAM... ¿cuál afirmación es incorrecta?

- a.  $t_C$  (tiempo de ciclo)  $< t_a$  (tiempo de acceso) ó  $t_{RAC}$ , porque hay que precargar la fila ( $t_{RP}$ ) tr.49, >
- b. En lectura, se proporciona la dirección de fila y se activa #RAS, luego la dirección de columna y se activa #CAS, y un cierto tiempo después ( $t_{RAC}$  tras #RAS, o  $t_{CAC}$  tras #CAS) el dato solicitado está disponible en el bus tr.49-52
- c. En lectura de dos palabras,  $t_{aa} = t_{RAC} + t_{RP} + t_{RAC}$ , porque hay que volver a proporcionar la fila de la segunda palabra
- d. En modo FPM, para leer dos palabras de la misma fila  $t_{aa} = t_{RAC} + t_{CP} + t_{CAC}$ , bastante menor que  $t_{aa}$  en c). tr.53

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.

Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | b | c | b | c |

1. Una SRAM de 32Kx8bit (256Kbit) puede venir organizada en 512 filas, dedicando por tanto al decodificador de columnas...
  - a. 6 bits
  - b. 7 bits
  - c. 8 bits
  - d. 9 bitsver Tema6, tr.58
2. Para construir una DRAM de 4MB con pastillas de 1Mx1bit, se pueden conectar...
  - a. las líneas D0-D8 del bus de datos a las patillas D0-D8 de todos los módulos
  - b. las líneas A0-A19 del bus de direcciones a las patillas A0-A19 de todos los módulos
  - c. las líneas A0-A9 y A10-A19 a un multiplexor que lleva a las patillas A0-A9 de todos los módulos ver Tema6, tr.80
  - d. las líneas A20-A22 al decodificador que activa los 8 módulos que se implementan la posición de memoria solicitada
3. Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta
  - a. 64 pastillas
  - b. 32 pastillas
  - c. 16 pastillas
  - d. 8 pastillas $2^{32} \times 8 = (2^{29} \times 4) \times (2^3 \times 2)$
4. En un sistema de memoria el espacio de memoria es 4GB, el tamaño de línea es 64B, y hay 1MB de cache asociativa por conjuntos de 4 vías. ¿Cuántos bits se dedican a la etiqueta en el directorio cache?
  - a. 12 bits
  - b. 14 bits
  - c. 18 bits
  - d. Ninguna de las respuestas anteriores es correcta $2^{20}/2^2=2^{18}/2^6=2^{12}$  conj.  $\rightarrow 32=14+12+6$
5. Suponiendo que existiera un sistema x86-64 con una cache de 4KB, asociativa por conjuntos de 4 vías, con 4 conjuntos, ¿cuántos enteros cabrían en una línea (marco) de cache?
  - a. 8
  - b. 16
  - c. 32
  - d. 64Tema6, tr. 135-138, 1KB/conj, 256B/línea, 32 ints

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | a | b | c | b |

**1.** En la etapa de Decodificación, la Unidad de Control utiliza el registro

- a. PC (contador de programa)
- b. MAR (Memory Address Register)
- c. MDR (Memory Data Register)
- d. IR (Instruction Register)

Ver Tema1, tr.26-28

**2.** Un dato con direccionamiento inmediato está almacenado...

- a. En la propia instrucción máquina, tras el codop (y posiblemente tras otra información de direccionamiento)
- b. En un registro del procesador, listo para ser usado inmediatamente por la ALU
- c. En memoria principal, junto con todos los datos que usa el programa
- d. En memoria virtual

Ver Tema1, tr.32-33

**3.** Un dispositivo de E/S con buffers es probable que también disponga de otra característica arquitectural, a saber...

- a. Línea RDY para insertar estados de espera
- b. Línea IRQ para petición de interrupción
- c. Línea BR para petición de uso del bus
- d. Bus de datos duplicado para transferir independientemente datos de E/S y datos del buffer

Ver Tema1 tr.39

**4.** Una máquina de tipo 2/3 corresponde a...

- a. Una máquina de pila
- b. Una máquina de acumulador
- c. Una máquina de registros de propósito general
- d. Ninguna de las respuestas anteriores es correcta

Ver Tema1 tr.19

**5.** Un benchmark de CPU intenta medir únicamente las prestaciones de la CPU, pero inevitablemente también mide...

- a. Las prestaciones del compilador y de los dispositivos de E/S
- b. Las prestaciones de la memoria cache y del bus del sistema (en la placa madre)
- c. Las prestaciones de la memoria principal y de la memoria virtual
- d. Todas las respuestas anteriores son correctas

Ver Tema1 tr.50

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | a | a | d | c |

1. En la arquitectura von Neumann
  - a. Se distinguen 5 componentes: Entrada, Salida, Memoria y CPU
  - b. El concepto fundamental es que el programa se almacena en memoria (*stored-program concept*), ya que las operaciones de la ALU son fijas (no se va a cambiar su diseño) y por tanto se pueden codificar
  - c. Los contenidos de memoria se interpretarán como instrucciones o datos (y los datos se interpretarán como caracteres, enteros, punto flotante, etc) según el contexto impuesto por el programa en ejecución
  - d. Todas las respuestas anteriores son correctas

Ver Tema1, tr.7-8.
2. Las instrucciones máquina de un computador digital se pueden clasificar...
  - a. En tres tipos: transferencia, operaciones, control
  - b. En cuatro tipos: aritméticas, lógicas, caracteres, punto flotante
  - c. En cinco tipos: uno para cada unidad funcional (Entrada, Salida, etc)
  - d. En principio cada computador tendrá el repertorio de su procesador, no se pueden establecer clasificaciones generales

Ver Tema1, tr.8.
3. La Entrada/Salida de un computador tiene como propósito comunicarse con, fundamentalmente...
  - a. Tres objetivos: un operador humano, un dispositivo de almacenamiento, u otro computador
  - b. Cuatro objetivos: tarjeta gráfica, disco duro, teclado y ratón
  - c. Cinco objetivos: uno por cada unidad funcional
  - d. Hay tantos posibles objetivos que no se pueden establecer clasificaciones generales
4. Una máquina de tipo 1/1 corresponde a...
  - a. Una arquitectura R/R
  - b. Una arquitectura R/M
  - c. Una arquitectura M/M
  - d. Ninguna de las respuestas anteriores es correcta

Ver Tema1 tr.19
5. Siendo **I** el nº de instrucciones de un programa benchmark, **T** su tiempo de ejecución, **E** el nº medio de etapas por instrucción, y **F** la frecuencia de reloj, la ecuación básica de rendimiento se expresa como...
  - a.  $I = T / EF$
  - b.  $F = TI / E$
  - c.  $T = EI / F$
  - d. Todas las respuestas anteriores son incorrectas

Ver Tema1 tr.51

**Nombre:****DNI:****Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | a | b | d | c |

**1.** Por x86-64 se entiende la misma arquitectura de repertorio (ISA) que...

- a. x86
- b. IA-32
- c. IA-64
- d. AMD64

[Ver Tema2.1, tr.11](#)

**2.** En la arquitectura IA-32 cada posición de memoria es de tamaño...

- a. 8bits (un byte)
- b. 16bits (por compatibilidad con el primer 8086)
- c. 32bits (el ancho de palabra, el mismo tamaño que los registros)
- d. La arquitectura IA-32 no fija un ancho de memoria, varía según el modelo del procesador

[Ver Tema2.1, tr.15](#)

**3.** La función que realiza el comando `gcc -c p1.s` se denomina...

- a. Compilar
- b. Ensamblar
- c. Enlazar
- d. Linkar

[Ver Práct2 Fig.1/Tema2.1 tr.16](#)

**4.** La arquitectura IA-32 contempla realizar operaciones con datos enteros de tamaño...

- a. 8bits
- b. 16bits
- c. 32bits
- d. Todas las respuestas anteriores son correctas

[Ver Tema2.1 tr.18](#)

**5.** En la arquitectura IA-32...

- a. Hay 32 registros de tamaño 32bits, no de otros tamaños
- b. Hay 8 registros de tamaño 8bits, 16 registros de tamaño 16bits y 32 registros de tamaño 32bits
- c. Hay 8 registros de cada tamaño: 8, 16, 32bits
- d. Cada registro tiene una versión de 32bits (`E_X`), de 16bits (`_X`) y de 8 bits (`_H, _L`)

[Ver Tema2.1 tr.26](#)

**Nombre:****DNI:****Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| b | c | a | c | d |

**1.** Los procesadores IPF tienen arquitectura de repertorio (ISA)...

- a. IA-32
- b. IA-64
- c. AMD64
- d. EM64T

**Ver Tema2.1, tr.9**

**2.** En la arquitectura IA-32 el registro contador de programa se denomina...

- a. PC
- b. RCP
- c. EIP
- d. PC32

**Ver Tema2.1 tr.15**

**3.** La función que realiza el comando `gcc -S p1.c` se denomina...

- a. Compilar
- b. Ensamblar
- c. Enlazar
- d. Linkar

**Ver Práct2 Fig.1/Tema2.1 tr.16**

**4.** La arquitectura IA-32 contempla realizar operaciones con datos en punto flotante de tamaño...

- a. 32bits
- b. 64bits e incluso >64bits
- c. Ambas respuestas son correctas
- d. Ambas respuestas son incorrectas

**Ver Tema2.1 tr.18**

**5.** En x86\_64...

- a. Hay 64 registros de tamaño 64bits, no de otros tamaños
- b. Hay 8 registros de tamaño 8bits, 16 de 16bits, 32 de 32bits, y 64 registros de 64bits
- c. Hay 8 registros de cada tamaño: 8, 16, 32, 64bits
- d. Cada registro tiene una versión de 64bits (RN), de 32bits (RNd), de 16bits (RNw) y de 8 bits (RNb/l)

**Ver Práct2 p.15/Tema2.1 tr.43**

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | a | b | c |

1. Si %edx contiene 0xf000 y %ecx contiene 0x0100, el direccionamiento 0x80(%ecx,%edx,2) se refiere a la posición...
  - a. 0xf182
  - b. 0xf280
  - c. 0x1e180
  - d. Ninguna de las respuestas anteriores es correctaTema2.2, tr.5
2. ¿Cuál de las siguientes secuencias de instrucciones multiplica %eax por 10?
  - a. leal(%eax,%eax,4), %eax  
sall \$2, %eaxsería \$1
  - b. imull \$0x10, %eaxsería \$10
  - c. addl %eax, %eax  
shll \$5, %eaxsería imull
  - d. Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo unaTema2.2, tr.6,8
3. ¿Qué combinación de flags aritmético-lógicos corresponde al código de condición b (below)?
  - a. CF
  - b. OF
  - c. CF xor OF
  - d. OF xor SFTema2.2, tr.23
4. Para traducir una asignación condicional (a=b?c:d;) de lenguaje C a lenguaje ensamblador, gcc puede que utilice...
  - a. Un salto condicional, según la condición expresada en el código Csería opuesta
  - b. Un salto condicional, según la condición opuesta a la del código C, y otro salto incondicionalTema2.2, tr.28-32
  - c. Una instrucción de movimiento condicional, pero sólo si la distribución Linux es de 64bitso con –march=i686
  - d. Una instrucción de movimiento condicional, pero sólo si el procesador es de 64bitso Pentium Pro...
5. Para traducir una construcción “do-while” de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente, además del código correspondiente al cálculo de la condición...
  - a. un salto condicional
  - b. un salto condicional y otro incondicional
  - c. una etiqueta y un salto condicionalTema2.2, tr.38-40
  - d. una etiqueta, un salto condicional y otro incondicional

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | d | c | b | b |

1. En los modos de direccionamiento del tipo Desplazamiento(Base,Indice,FactorEscala), puede usarse como... Tema2.2, tr.4
  - a. desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)
  - b. base, cualquiera de los 8 registros enteros salvo %esp sería cualquiera
  - c. índice, también cualquiera salvo %ebp sería %esp
  - d. factor de escala, cualquier constante de 1, 2, 4 u 8 bytes sin bytes
2. ¿Cuál de las siguientes secuencias de instrucciones calcula a=b-a, suponiendo que %eax es a y %ebx es b?
  - a. subl %eax, %ebx b=b-a
  - b. subl %ebx, %eax a=a-b
  - c. notl %eax  
addl %ebx, %eax sería negl
  - d. Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una Tema2.2, tr.9
3. La diferencia entre las instrucciones test y cmp consiste en que...
  - a. test realiza una operación and lógico, mientras que cmp realiza una resta
  - b. test modifica sólo los flags lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
  - c. ambas respuestas son correctas Tema2.2, tr.21-22
  - d. ambas respuestas son incorrectas
4. Para traducir una construcción if-then-else de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente...
  - a. Un salto condicional, según la condición expresada en el código C
  - b. Un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional Tema2.2, tr.28-32
  - c. Dos saltos condicionales (uno para la parte if y otro para la parte else)
  - d. Dos saltos condicionales y dos saltos incondicionales
5. Para traducir una construcción “do-while” de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente ...
  - a. un salto condicional hacia adelante, según la misma condición que en lenguaje C
  - b. un salto condicional hacia atrás, según la misma condición que en lenguaje C Tema2.2, tr.38-40
  - c. un salto condicional hacia adelante, según la condición opuesta a la de lenguaje C
  - d. un salto condicional hacia atrás, según la condición opuesta a la de lenguaje C

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | c | a | a | c |

1. Para desplazar %eax a la derecha un número variable de posiciones, indicado en %ebx, se puede hacer...
  - a. sar %ebx, %eax no existe
  - b. sar %bl, %eax no existe
  - c. mov %ebx, %ecx p.213 libro, Ej.3.8, p.214  
sar %cl, %eax
  - d. No se puede, sar sólo admite un número fijo de posiciones (debe ser un valor inmediato)
2. Indicar cuál es la dirección de salto (en qué dirección empieza la subrutina <main>) para esta instrucción call  
0804854e: e8 3d 06 00 00      call <main>  
08048553: 50                    pushl %eax
  - a. 0804854e + 3d060000 no, offset little-endian
  - b. 0804854e + 0000063d = 08048b8b no, %eip=08048553
  - c. 08048b90 Tema2.3, tr.7
  - d. No se puede deducir de la información proporcionada, faltan datos
3. El motivo para usar una estructura de pila para almacenar las direcciones de retorno es...
  - a. que, obviamente, como no puede ser de otra manera, se debe retornar de la subrutina invocada antes que de la invocante, con lo cual se consume la última dirección de retorno anotada antes que la anterior Tema2.3, tr.10
  - b. que así se pueden almacenar argumentos y variables locales, cosa que no se podría hacer sin usar una pila (no serviría una cola, ni un montículo (heap), ni por supuesto los registros de la CPU)
  - c. que así se pueden gestionar funciones recursivas y reentrantes, además de funciones anidadas “normales” (no reentrant o recursivas) que sí podrían haberse implementado con otros mecanismos distintos a una pila
  - d. elección del fabricante, puesto que existen fabricantes que usan colas, otros usan montículos, otros usan registros de la CPU, y todos pueden gestionar funciones “normales”, anidadas, recursivas y reentrantes con la misma facilidad
4. Considerando que un marco de pila va desde una dirección de retorno a la siguiente, los argumentos de llamada a una función...
  - a. están en el marco de pila del invocante, y es responsabilidad del invocante ponerlos al llamar Tema2.3, tr.24
  - b. están en el marco de pila del invocante, y es responsabilidad del invocado ponerlos al llamar
  - c. están en el marco de pila del invocado, y es responsabilidad del invocante ponerlos al llamar
  - d. están en el marco de pila del invocado, y es responsabilidad del invocado ponerlos al llamar
5. En la convención de llamada cdecl seguida por gcc Linux/IA-32...
  - a. EAX es un registro salva-invocante, por eso en cualquier función hay que salvarlo antes de usarlo no para usarlo
  - b. EBX es un registro salva-invocante, por eso si se usa hay que salvarlo antes de llamar a una función salva-invocado
  - c. ESI es un registro salva-invocado, por eso en cualquier función hay que salvarlo antes de usarlo Tema2.3, tr.36
  - d. EDI es un registro salva-invocado, por eso si se usa hay que salvarlo antes de llamar a una función no para llamar

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | a | c | b |

**1.** La instrucción push \$0xff...

- a. Equivale a pushb \$0xff, mete un byte en la pila
- b. Equivale a pushw \$0x00ff, mete 2 bytes en la pila
- c. Equivale a pushl \$0x000000ff, mete 4 bytes en la pila
- d. Da un error de ensamblado, hace falta indicar el tamaño del operando usando b/w/l

error, no existe esa instr.  
eso existe, pero no equivale  
Ej.3.2.5, p.208 libro  
no da error, asume pushl

**2.** Indicar cuál es la dirección de la instrucción mov en el siguiente desensamblado, donde se ha borrado parte de la dirección

0804xxxx: 74 12 je 08048391  
0804xxxx: b8 00 00 00 00 mov \$0, %eax

- a. 08048391 + 12 = 08048403
- b. 08048391 – 12 = 08048379
- c. 0804837d
- d. 0804837f

Ej.3.3.15c, p.226 libro y Tema2.3, tr.7

**3.** Respecto a funciones recursivas y reentrantes...

- a. Toda función recursiva es reentrant, aunque no toda función reentrant es recursiva
- b. Toda función reentrant es recursiva, aunque no toda función recursiva es reentrant
- c. Ambos calificativos son sinónimos, se prefiere llamarlas “funciones recursivas” en lenguajes de alto nivel, y “subrutinas reentrantes” en el contexto de lenguaje ensamblador, contadores de programa y direcciones de retorno
- d. Los dos calificativos no tienen nada que ver, son características independientes (ortogonales), pueden ponerse ejemplos de funciones que sean de un tipo, del otro, de ambos y de ninguno

Tema2.3, tr.10

**4.** El ajuste de marco de pila que gcc (Linux/IA-32) prepara para todas las funciones consiste en las instrucciones...

- a. movl %ebp, %esp  
pushl %ebp
- b. movl %esp, %ebp  
pushl %esp
- c. pushl %ebp  
movl %esp, %ebp
- d. pushl %esp  
movl %ebp, %esp

Tema2.3, tr.28

**5.** Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/IA-32 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como...

- a. var (el nombre de la variable representa su posición de memoria)
- b. -k(%ebp), siendo k un número constante relativamente pequeño
- c. k(%ebp), siendo k un número constante relativamente pequeño
- d. k(%esp), siendo k un número constante relativamente pequeño

Tema2.3, tr.46-48

**Nombre:**

DNI:

## Grupo:

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea. Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | b | d | d | d |

1. En x86\_64 se pueden referenciar los registros... p.202 libro (F.3.2), p.307-8 (F.3.35), Prácticas Ap.2 p.15, T2.4 tr.4, T2.1 tr.26,43

  - a. %rax, %eax, %ax, %ab %ab no existe
  - b. %rsi, %esi, %si, %sib %sib no existe
  - c. %r8q, %r8d, %r8w, %r8b %r8q no existe, es %r8 a secas
  - d. %r12, %r12d, %r12w, %r12b ok

2. En general, gcc Linux x86\_64 evitará crear un marco de pila tanto en el invocante como en el invocado, a menos que por algún motivo no se pueda evitar. Alguno de los siguientes motivos no es inevitable:

  - a. Realizar en el invocante una llamada con más de 6 argumentos ok, T2.4 tr.7, <=6 -> regs
  - b. Usar en el invocado una variable local declarada con la palabra reservada volatile no, T2.4 tr.8, usa zona roja
  - c. Necesitar usar (conservar) un registro salva-invocado mientras se llama a otra subrutina ok, T2.4 tr.10, %rsp baja
  - d. Necesitar usar (conservar) un registro salva-invocante mientras se llama a otra subrutina ok, mismo motivo

3. Si declaramos int val[5]={1,5,2,1,3}; entonces...

  - a. val[5] es de tipo int y vale 3 T2.4, tr.18
  - b. val+1 es de tipo int y vale 2 no, val[4]==3
  - c. val+4 es de tipo int\* y se cumple que \*(val+4)==5 no, int\*, val+1==&val[1]
  - d. &val[2] es de tipo int\* y vale lo mismo que val+8 no, \*(val+4)==val[4]==3

4. Una función C llamada get\_el() genera el siguiente código ensamblador. Se puede adivinar que...

```
movl 8(%ebp), %eax
leal (%eax,%eax,4), %eax
addl 12(%ebp), %eax
movl var(,%eax,4), %eax
```

  - a. var es un array multi-nivel (punteros a enteros) de cuatro filas
  - b. var es un array multi-nivel pero no se pueden adivinar las dimensiones
  - c. var es un array bidimensional de enteros, no se pueden adivinar dimensiones
  - d. var es un array bidimensional de enteros, con cinco columnas ok, T2.4 tr.28

5. Al traducir la sentencia C r->i = val; gcc genera el código ASM movl %edx, 12(%eax). Se deduce que...

  - a. r es un puntero que apunta a la posición de memoria 12
  - b. i es un entero que vale 12
  - c. val es un entero que vale 12
  - d. el desplazamiento de i en \*r es 12 ok, T2.4 tr.40

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | b | d | b | b |

1. En x86\_64 se pueden referenciar los registros... p.202 libro (F.3.2), p.307-8 (F.3.35), Prácticas Ap.2 p.15, T2.4 tr.4, T2.1 tr.26,43
  - a. %rax, %eax, %ax, %ah, %al siempre que no vayan %ah y uno de los nuevos (p.ej. %sil) en la misma instrucción
  - b. %rsi, %esi, %si, %sih, %sil %sih no existe
  - c. %r8, %r8d, %r8w, %r8l %r8l no existe, es %r8b
  - d. %r12q, %r12d, %r12w, %r12l ni %r12l ni %r12q, es %r12 a secas
2. Comparando las convenciones de llamada de gcc Linux IA-32 con x86\_64 respecto a registros... T2.3 tr.36, T2.4 tr.5
  - a. En IA-32 %ebx es salva-invocante, pero en x86\_64 %rbx es salva-invocado ambos salva-invocado
  - b. En IA-32 %ecx es salva-invocante, y en x86\_64 %rcx es salva-invocante también ok
  - c. En IA-32 %esi es salva-invocado, y en x86\_64 %rsi es salva-invocado también en x86\_64 salva-invocante
  - d. En IA-32 %ebp es especial (marco de pila), y en x86\_64 %rbp tambien en x86\_64 salva-invocado
3. Si definimos int vec\_ele(int n, int \*a, int i, int j){ return a[n\*i+j];}, entonces...
  - a. Se puede llamar a la función con cualquier vector a que tenga al menos  $n*i+j$  elementos no,  $(n*i+j)+1$  elementos
  - b. Se espera que a sea un array bidimensional de n filas y n columnas
  - c. Se espera que a sea un array bidimensional de n filas, i es la fila y j la columna
  - d. Se espera que a sea un array bidimensional de n columnas ok, T2.4 tr.32
4. Una función C llamada get\_el() genera el siguiente código ensamblador. Se puede adivinar que...

```
movl 8(%ebp), %eax
movl var(,%eax,4), %edx
movl 12(%ebp), %eax
movl (%edx,%eax,4), %eax
```

  - a. var es un array multi-nivel (punteros a enteros) de cuatro filas
  - b. var es un array multi-nivel pero no se pueden adivinar las dimensiones ok, T2.4 tr.30
  - c. var es un array bidimensional de enteros, no se pueden adivinar dimensiones
  - d. var es un array bidimensional de enteros, con cinco columnas
5. Al traducir la sentencia C `return &r->a[idx];` gcc genera el código ASM siguiente. Se deduce que...

```
movl 12(%ebp), %eax
sall $2, %eax
addl 8(%ebp), %eax
```

  - a. r es una estructura, y su dirección (&r) es 12(%ebp)
  - b. a es un array y es el primer campo de la estructura ok, T2.4 tr.41
  - c. idx vale 12
  - d. idx vale 4

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | a | d | d | d |

1. La diferencia entre “unidad aritmético-lógica” y “unidad de procesamiento” consiste en que:
  - a. La primera está incluida en la segunda
  - b. La segunda está incluida en la primera
  - c. Son unidades distintas, aunque relacionadas
  - d. Son distintos nombres para la misma unidadTema3, tr.6
2. Es típico que los registros del banco de registros tengan líneas de control como:
  - a. “Load” y “Enable”, en unidades de control con un bus
  - b. “LoadB”, “EnableB”, “LoadM” y “EnableM”, en unidades con un bus, según se lea/escriba al bus interno o a memoria
  - c. “LoadB”, “EnableB”, “LoadM” y “EnableM”, en unidades con buses múltiples, según se lea/escriba al bus interno o a memoria
  - d. “Load[A,B,C]” y “Enable[A,B,C]”, por ejemplo (6 en total), en unidades con 3 buses A,B,C, según de qué bus se lea/escribaTema3, tr.13,19,21
3. Alguna de las siguientes señales \*no\* sirve de entrada a la unidad de control. ¿Cuál?
  - a. Señal de reloj (CLK)
  - b. Estado de la unidad de proceso (flags Z,S,C,O...)
  - c. Instrucción actual (bits del registro IR)
  - d. Contador de programa (bits del registro PC)Tema3, tr.25
4. Respecto a los términos microinstrucción y microcódigo:
  - a. Son equivalentes, llamamos microcódigo o microinstrucción a una palabra de la memoria de control
  - b. Una microinstrucción está programada en microcódigo, que es un lenguaje para programar señales de control
  - c. Un microcódigo controla una serie de señales de control relacionadas (por ejemplo, el código 000 para que la ALU realice la suma), y varios microcódigos juntos forman una microinstrucción
  - d. Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoriaTema3, tr.42, d
5. Parecidos y diferencias entre secuenciamiento explícito e implícito en una UC microprogramada. Alguna de las siguientes afirmaciones \*no\* es correcta.
  - a. En el implícito hay un microcontador de programa, en el explícito no
  - b. En el explícito hay un campo de dirección de la siguiente microinstrucción (o incluso dos, para microbifurcaciones), en el implícito no hay tal campo (o sólo uno, para microbifurcaciones, por lo que se suele solaparlo con otros campos)
  - c. En el implícito se necesita una ROM o PLA de traducción codop→dirección 1ª microinstrucción, en el explícito no
  - d. Ambos secuenciamientos necesitan un campo de 1 bit “TS” (tipo de salto) con el cual se marca la 1ª microinstrucción de una instrucción máquina

Sería la última uinstr, no la 1ª. Tema3, tr.53-58

Después veremos el ej. de Tanenbaum (tr. 82,84,86,89) que no tiene TS.

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | d | b | d |

1. La ganancia en velocidad ideal de un cauce de K etapas de igual duración ejecutando un programa de N instrucciones es:
  - a.  $S = KN/(K-N+1)$
  - b.  $S = NKT/(N-K+1)T$
  - c.  $S = KN/(K+N-1)$
  - d.  $S = NT/(N+K-1)T$Tema4, p.18
2. Alguno de los siguientes \*no\* es un motivo de que no se alcance la ganancia ideal en un cauce segmentado
  - a. El propio coste de la segmentación (carga de los registros de acople, etc...)
  - b. La duración del ciclo de reloj impuesta por la etapa más lenta
  - c. Los riesgos (*hazards*)
  - d. La emisión múltiple (y posiblemente desordenada) de instruccionesTema 4, tr.19
3. Alguno de los siguientes \*no\* es un riesgo estructural
 (d es de datos) Tema 4, tr.24-27,28-29
  - a. Etapa M(emory acc.) simultánea con etapa F(etch) en una arquitectura von Neumann (no Harvard)
  - b. Instrucción que emplea varios ciclos en etapa E(xecution), cuando las etapas previas (F,D) suelen durar 1 ciclo
  - c. Fallo de cache al captar una instrucción (en ausencia de cola de instrucciones)
  - d. Instrucción que requiere el valor de un registro (en etapa Decode) que se está modificando por la instrucción anterior (aún no ha llegado a etapa WriteBack), en ausencia de *bypass* (*register forwarding*)
4. La técnica de *register forwarding* (*bypass*) está relacionada con...
  - a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
  - b. Los riesgos de datos (intenta que el dato esté disponible anticipadamente)
  - c. Los riesgos de control (intenta determinar de antemano el flujo de control)
  - d. Los riesgos de direcciones (intenta no acceder a la dirección mientras se está captando una instrucción)Tema 4, tr.29
5. Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?
  - a. El retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)
  - b. El retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple
  - c. El retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca
  - d. El retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de saltoTema 4, tr.37-38

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | c | b | b | d |

1. Respeto a los conceptos de procesador de E/S, canal de E/S, dispositivos de E/S:
  - a. Un procesador o canal tiene un repertorio de instrucciones específico para manejar los dispositivos E/S **ver T5 tr.5**
  - b. Cada canal es una línea de comunicación entre el procesador y un dispositivo de E/S.
  - c. Al conjunto de conexiones entre el procesador y los dispositivos se le denomina canal de E/S (de ese ordenador)
  - d. La pregunta es capciosa, el procesador no es E/S, son otros dos componentes von Neumann distintos (ALU+UC)
2. Respeto a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida:
  - a. Todas ellas impiden que haya cortocircuito en el bus de datos
  - b. Todas ellas impiden que haya cortocircuito en el bus de direcciones
  - c. La selección lineal permitiría escribir un mismo dato a varios puertos E/S **ver Tema5 tr.7-9**
  - d. Usando decodificación centralizada es más fácil realizar expansiones al sistema de E/S
3. Respeto a las operaciones de lectura y escritura, con temporización síncrona o asíncrona: **ver Tema5 tr.11-16**
  - a. En lectura síncrona, el procesador pone dirección y datos en los buses correspondientes, y espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado **datos no (ACK tampoco)**
  - b. En escritura síncrona, el procesador pone dirección y datos en los buses correspondientes (y también los retira de los buses) en instantes de tiempo determinados por el reloj del bus **ok**
  - c. En lectura asíncrona, el procesador pone los datos en el bus de datos, y espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado **datos no**
  - d. En escritura asíncrona, el procesador pone la dirección en el bus de direcciones (y también la retira) en instantes de tiempo determinados por el reloj del bus **reloj bus no**
4. Respeto a si un computador dispone de E/S independiente (separada) y/o usa E/S mapeada a memoria: **Tema5 tr.21-28**
  - a. Si el encapsulado (chip) del procesador tiene patilla (pin) IO/M# (o patillas equivalentes), eso evidencia que el computador usa E/S mapeada a memoria **dispone de E/S separada, otra cosa es que se use o no**
  - b. Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada
  - c. Si el encapsulado del procesador no dispone de patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada **E/S mapeada siempre disponible, otra cosa es si se usa o no**
  - d. Si el repertorio del procesador tiene instrucciones del tipo LOAD y STORE, el computador sólo dispone de E/S mapeada a memoria **puede tener además IN y OUT para E/S separada**
5. Alguna de las siguientes \*no\* es una ventaja de la E/S independiente (separada, aislada)
  - a. Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
  - b. Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
  - c. Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
  - d. Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño) **ver Tema5, tr.23,25**

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| b | d | a | c | b |

1. Suponer que el tiempo (T) de ejecución de una instrucción en un procesador pudiera dividirse en 4 etapas, pero no de igual duración, sino p.ej.: T/3, T/5, T/4, T/4. ¿Cuál sería aproximadamente la ganancia (S) del correspondiente cauce segmentado?
  - a. No se puede hacer un cauce segmentado con etapas de distinta duración
  - b. Depende de la etapa más lenta, y en este caso sería 3
  - c. Depende de la etapa más rápida, y en este caso sería 5
  - d. Depende del número de etapas, y en este caso sería 4Tema4, pp.18-19
2. Alguno de los siguientes \*no\* es un riesgo que hayamos estudiado en cauces segmentados
  - a. Riesgos estructurales
  - b. Riesgos de (dependencia de) datos
  - c. Riesgos de control
  - d. Riesgos de transferenciaTema 4, tr.24-27
  
Tema 4, tr.28-30
  
Tema 4, tr.31-35
  
inventado
3. La precaptación (cola de instrucciones) está relacionada con...
  - a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
  - b. Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
  - c. Los riesgos de control (intenta determinar de antemano el flujo de control)
  - d. Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)Tema 4, tr.27
4. La predicción de saltos está relacionada con...
  - a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
  - b. Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
  - c. Los riesgos de control (intenta determinar de antemano el flujo de control)
  - d. Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)Tema 4, tr.39-40
5. ¿Cuál de los siguientes modos de direccionamiento es \*menos\* preferible para un procesador con segmentación de cauce?
  - a. Registro
  - b. Directo
  - c. Indirecto a través de registro
  - d. Indexado (o relativo a base, o base+índice)Tema 4, tr.42

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| a | d | d | c | b |

1. ResPECTO a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:
  - a. La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo ver Tema5 tr.5-6
  - b. Cada puerto o interfaz es una línea de comunicación con el procesador. El conjunto de ellos forma el controlador.
  - c. El puerto, o interfaz, contiene los controladores necesarios para comunicar el dispositivo con el procesador
  - d. El interfaz contiene las controladoras necesarias para conectar los puertos con el procesador
2. ResPECTO a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida:
  - a. La selección lineal permitiría leer simultáneamente un dato desde varios puertos E/S
  - b. La decodificación centralizada permitiría escribir simultáneamente un dato a varios puertos E/S
  - c. Usando decodificación centralizada es más fácil realizar expansiones al sistema de E/S
  - d. Usando decodificación distribuida cada dispositivo debe incluir hardware para fijar y reconocer su dirección v.T5 tr.7-9
3. ResPECTO a las operaciones de lectura y escritura, con temporización síncrona o asíncrona:
 ver Tema5 tr.11-16
  - a. En lectura síncrona, el procesador pone dirección y datos en los buses correspondientes en instantes de tiempo determinados por el reloj del bus datos no
  - b. En escritura síncrona, el procesador pone la dirección en el bus de direcciones, espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado, lee los datos y retira la dirección sería \*lectura\* asíncrona
  - c. En lectura asíncrona, el procesador pone la dirección en el bus de direcciones, y posteriormente lee del bus de datos, ambas acciones en instantes de tiempo determinados por el reloj del bus sería lectura \*síncrona\*
  - d. En escritura asíncrona, el procesador pone dirección y datos en los buses correspondientes, y espera a que se active una señal de conformidad ACK por parte del dispositivo direccionado ok
4. Si nos dicen que en un sistema computador cada dirección especifica uno o dos puertos de E/S, se refiere a que:
  - a. La misma dirección (por ejemplo 0x0210) puede ser una posición de memoria o un puerto de E/S, según IO/M#
  - b. La misma dirección puede usarse para transferir un byte o una palabra de mayor tamaño (ese byte y el siguiente)
  - c. Un puerto será de sólo lectura, otro de sólo escritura, y ambos se decodifican en la misma dirección ver Tema5 tr.6
  - d. La pregunta es capciosa, una dirección puede especificar un puerto, no dos
5. Alguna de las siguientes no es una técnica de E/S de las estudiadas en clase:
  - a. E/S programada
  - b. E/S asíncrona ver T5 tr.3,30-32
  - c. E/S controlada por interrupciones
  - d. E/S mediante Acceso Directo a Memoria

**Nombre:**

**DNI:**

**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | b | d | b |

1. Parecidos y diferencias entre los métodos de E/S: Alguna de las siguientes afirmaciones es **incorrecta**: T5 tr.29-32,37,58,43,44
  - a. La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa *polling* para determinar el origen de la IRQ)
  - b. Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó)
  - c. Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S **E/S IRQ compartida dispTvs**
  - d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
2. Tipos de interrupción que suelen contemplar las CPUs comerciales actuales: Alguno de los siguientes **\*no\*** lo es:
  - a. Internas (excepciones o *traps*): generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, fallo de página, etc)
  - b. Externas (IRQs hardware): generadas por un dispositivo externo a la CPU, activan la línea #INTR (o equivalente)
  - c. Software: generadas al ejecutar la instrucción INT (o equivalente)
  - d. Firmware (*faults*): generadas por el microcódigo de la CPU (*segm. fault, page fault...*) **ver Tema5 tr.50-51**
3. Alguna de las siguientes técnicas **\*no\*** es de utilidad para determinar la causa de una interrupción **ver Tema5 tr.56-58**
  - a. Múltiples líneas de interrupción INT1#, INT2#...
  - b. Línea de reconocimiento INTA# **básica para retirar IRQ, evitar redispacho, enviar vector, T5 tr.46,55**
  - c. Consulta de estado, o *polling*
  - d. Interrupciones vectorizadas
4. Alguna de las siguientes técnicas **\*no\*** está relacionada con la gestión por prioridad de IRQs simultáneas:
  - a. Centralizada : un único circuito recibe las diversas peticiones y reconoce la prioritaria
  - b. Sondeo, o *polling*: un software (la ISR) sondea el estado, la prioridad viene establecida por el orden de consulta
  - c. Encadenamiento, o *daisy-chain*: del reconocimiento INTA#, la prioridad viene establecida por el orden en la cadena
  - d. Inhibición: un registro de máscara retiene las peticiones menos prioritarias hasta que les llegue el turno **T5 tr.60-62,68**
5. Respeto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR):
  - a. No es necesario salvar ninguno más, el contador de programa y los flags de estado ya los salva la propia CPU como parte del mecanismo de interrupción
  - b. Se deben guardar los registros que se modifiquen en la propia ISR. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar **ver Tema5, tr.46**
  - c. Se deben guardar los registros salva-invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA-32), los registros salva-invocante ya los guarda el programa interrumpido
  - d. Se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción

**Nombre:****DNI:****Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| c | d | b | d | b |

1. Parecidos y diferencias entre los métodos de E/S: Alguna de las siguientes afirmaciones es **incorrecta**: T5 tr.29-32,37,58,43,44
  - a. La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa *polling* para determinar el origen de la IRQ)
  - b. Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó)
  - c. Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S   **E/S IRQ compartida dispTvs**
  - d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
2. Tipos de interrupción que suelen contemplar las CPUs comerciales actuales: Alguno de los siguientes **\*no\*** lo es:
  - a. Internas (excepciones o *traps*): generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, fallo de página, etc)
  - b. Externas (IRQs hardware): generadas por un dispositivo externo a la CPU, activan la línea #INTR (o equivalente)
  - c. Software: generadas al ejecutar la instrucción INT (o equivalente)
  - d. Firmware (*faults*): generadas por el microcódigo de la CPU (*segm. fault, page fault...*)      **ver Tema5 tr.50-51**
3. Alguna de las siguientes técnicas **\*no\*** es de utilidad para determinar la causa de una interrupción   **ver Tema5 tr.56-58**
  - a. Múltiples líneas de interrupción INT1#, INT2#...
  - b. Línea de reconocimiento INTA#      **básica para retirar IRQ, evitar redispacho, enviar vector, T5 tr.46,55**
  - c. Consulta de estado, o *polling*
  - d. Interrupciones vectorizadas
4. Alguna de las siguientes técnicas **\*no\*** está relacionada con la gestión por prioridad de IRQs simultáneas:
  - a. Centralizada : un único circuito recibe las diversas peticiones y reconoce la prioritaria
  - b. Sondeo, o *polling*: un software (la ISR) sondea el estado, la prioridad viene establecida por el orden de consulta
  - c. Encadenamiento, o *daisy-chain*: del reconocimiento INTA#, la prioridad viene establecida por el orden en la cadena
  - d. Inhibición: un registro de máscara retiene las peticiones menos prioritarias hasta que les llegue el turno **T5 tr.60-62,68**
5. Respeto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR):
  - a. No es necesario salvar ninguno más, el contador de programa y los flags de estado ya los salva la propia CPU como parte del mecanismo de interrupción
  - b. Se deben guardar los registros que se modifiquen en la propia ISR. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar      **ver Tema5, tr.46**
  - c. Se deben guardar los registros salva-invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA-32), los registros salva-invocante ya los guarda el programa interrumpido
  - d. Se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción

**Nombre:**
**DNI:**
**Grupo:**

Sobre 10, cada respuesta vale 2 si es correcta, 0 si está en blanco o claramente tachada, y -2/3 si es errónea.  
 Anotar las respuestas (**a**, **b**, **c** o **d**) en la siguiente tabla.

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| d | c | b | c | b |

1. Parecidos y diferencias entre los métodos de E/S: Alguna de las siguientes afirmaciones es **incorrecta**: [T5 tr.29-32,37,58,43,44](#)
  - a. En E/S programada la CPU debe encargarse tanto de consultar el estado del dispositivo de E/S como de transferir el dato deseado. También hay dispositivos que siempre están listos para transferir y no requieren consulta de estado.
  - b. En E/S por IRQ la CPU se ahorra la consulta de estado, pero sigue teniendo que transferir el dato. También hay formas de conexión de las líneas IRQ que requieren que la CPU haga consulta de estado.
  - c. En E/S por DMA la CPU se ahorra tanto la consulta de estado como la transferencia de datos. Pero se usan instrucciones de E/S para programar el controlador DMA, y la CPU suele recibir peticiones de interrupción (IRQ) del propio DMA.
  - d. Todos los métodos de E/S pueden terminar usando (opcionalmente) alguna característica de otro método (E/S DMA usando IRQs, E/S IRQs usando *polling*, etc) [E/S prog. no usa IRQ-DMA](#)
2. Tipos de interrupción que suelen contemplar las CPUs comerciales actuales: Alguno de los siguientes **\*no\*** lo es:
  - a. Software: generadas al ejecutar la instrucción INT (o equivalente)
  - b. Hardware (IRQs externas): generadas por un dispositivo externo a la CPU, activan la línea #INTR (o equivalente)
  - c. Faltas o *faults*: generadas por el SO (*segment.fault*, *page fault*, *mem.prot.fault...*) [ver Tema5 tr.50-51](#)
  - d. Excepciones o *traps*: generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, *overflow*, etc)
3. Alguna de las siguientes técnicas **\*no\*** es de utilidad para determinar la causa de una interrupción [ver Tema5 tr.56-58](#)
  - a. Consulta de estado, o *polling*
  - b. *Daisy-chain* (encadenamiento) de ACK# [útil para prioridad entre IRQs simultáneas, ver Tema5 tr.65-67](#)
  - c. Interrupciones vectorizadas
  - d. Múltiples líneas de interrupción INT1#, INT2#...
4. En un sistema computador con varios dispositivos que hagan IRQ aparecen algunos problemas que se suelen resolver mediante mecanismos de prioridad. Alguno de los siguientes problemas o técnicas **\*no\*** está relacionado con este escenario:
  - a. Interrupciones simultáneas
  - b. Interrupciones anidadas
  - c. Redisparo de interrupciones [ver Tema5 tr.59](#)
  - d. Inhibición de interrupciones
5. Alguno de los siguientes **\*no\*** es un método de inhibición de interrupciones que hayamos estudiado en clase:
  - a. Mecanismos automáticos: como no atender interrupciones durante la primera instrucción de la ISR, o durante toda la ISR, o como no atender interrupciones de nivel igual o inferior a la prioridad que se está atendiendo actualmente
  - b. Mecanismos híbridos: combinación de esquemas centralizado y distribuido (*daisy-chain*) [ver Tema5 tr.70-72, 62](#)
  - c. Registro de máscara: la CPU (o controlador de IRQ) posee un registro en el que cada bit bloquea un nivel de interrupción, pudiendo desactivar selectivamente los niveles que no se desee atender
  - d. Flag general de inhibición: la CPU posee un flag y un par de instrucciones para activarlo y desactivarlo

# TEST EC PRACTICA 1

**1. ¿Cuál de los siguientes fragmentos es correcto para comenzar un programa en ensamblador que conste de un solo archivo.s?**

-> a) .section .text

```
.global _start
_start:
```

b) .text:

```
_start:
```

añadido (:) a (.text) para evitar subterfugios:

a) puede que .global \_start se añada más tarde (y funciona)

b) aunque no se ponga .global, ld supone dirección por defecto (y funciona)

Añadiendo (:) es error seguro (.text already defined)

c) section .text

```
.start _global
```

```
_start:
```

d) .section .text

```
.local _start
```

```
_start:
```

**2. El lenguaje máquina...**

a) es un conjunto de nombres simbólicos o nemotécnicos.

b) facilita la portabilidad de los programas.

c) es el mismo para todos los computadores.

-> d) Ninguna de las respuestas anteriores es correcta.

**3. Alguna de las siguientes líneas de código sirve para definir una variable entera llamada tam en GNU/as Linux x86. ¿Cuál?**

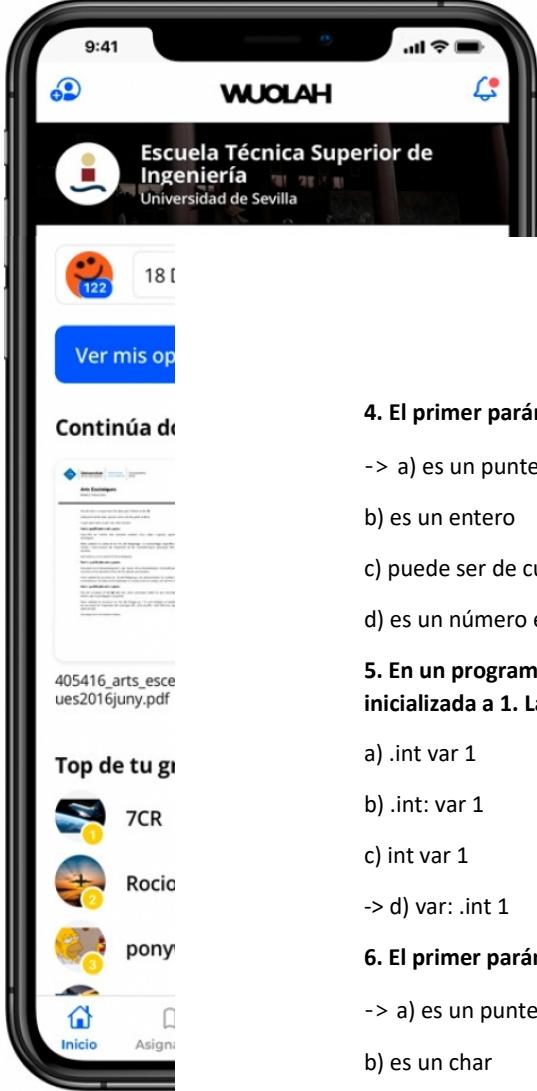
a) int tam;

El enunciado original decía "gcc/as" e introducía una ambigüedad indeseada con esta respuesta

b) var tam : integer;

-> c) tam: .int .-msg

d) \_int tam = 0



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

#### 4. El primer parámetro de printf:

- > a) es un puntero
- b) es un entero
- c) puede ser de cualquier tipo
- d) es un número en coma flotante

#### 5. En un programa en ensamblador queremos crear espacio para una variable entera var inicializada a 1. La línea que hemos de escribir en la sección de datos es:

- a) .int var 1
- b) .int: var 1
- c) int var 1
- > d) var: .int 1

#### 6. El primer parámetro de printf:

- > a) es un puntero
- b) es un char
- c) puede ser de cualquier tipo, incluso no existir
- d) es un entero

#### 7. El punto de entrada de un programa ensamblador en GNU/as Linux x86 se llama

- a) main
- > b) \_start
- c) \_init
- d) begin

#### 8. La etiqueta del punto de entrada a un programa ensamblador en el entorno de las prácticas 1 a 4 (GNU/as Linux x86) es:

- a) \_main
- > b) \_start

P3 y P4 se redactan en C.

P1 y P2 sí son en ensamblador GNU/as Linux x86.

Incluso en P2 se llega a usar main para ensamblar con gcc ya que usamos printf.

En cualquier caso, las otras opciones son descabelladas

- c) \_init
- d) .L0

**9. ¿De qué tipo son los procesadores Intel que usamos en los laboratorios?**

- a) el concepto de endian no es aplicable a estas máquinas, ya que un registro del procesador no cabe en una posición de memoria
  - b) big-endian
  - c) puede ajustarse mediante un bit de control en el registro CR0 que funcionen como little- o big-endian
- > d) little-endian

**10. Sobre el programa ensamblador:**

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
  - b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
  - c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.
- > d) Todas las respuestas son ciertas.

# TEST EC PRACTICA 2

**1. Si ECX vale 0, la instrucción adc \$0,%ecx**

a) Pone CF=1

-> b) Pone CF=0

c) Cambia CF

d) No cambia CF

**2. Alguno de los siguientes no es un nombre de registro en una máquina x86-64 en modo 64 bits**

a) r8d

b) r12w

-> c) sih

%sil en todo caso, no %sih

Ver libro Hallaron Figura 3.35

d) spl

**3. Alguno de los siguientes no es un nombre de registro en una máquina IA32 en modo 32 bits**

a) ebp

b) ax

c) dh

-> d) sil

Sí lo sería en modo 64 bits

Ver libro Hallaron Figura 3.35

**4. En el contexto de una llamada a función cdecl: si los contenidos de ESP y EBP son, respectivamente, 0x0008d040 y 0x00000000 antes de ejecutar una instrucción call, tras ejecutar el ret correspondiente, los contenidos serán:**

a) 0x0008d044 y 0x00000000

b) 0x0008d03c y 0x0008d03c

c) 0x0008d040 y 0x0008d040

-> d) 0x0008d040 y 0x00000000

**5. ¿Qué valor contendrá %edx tras ejecutar las siguientes instrucciones?**

xor %eax, %eax

sub \$1, %eax

cltd

idiv %eax

a) no puede saberse con los datos del enunciado

b) -1

c) 1

-> d) 0

**6. ¿Cuál de las siguientes no es una sección de un fichero ELF?**

a) .bss

para datos sin inicialización, mencionada en P2 Apéndice 2 Tabla 9

b) .data

para datos inicializados como en la Práctica "media"

-> c) .static

inventada

d) .text

para el código

**7. En contraposición a un ejecutable Linux ELF, un fichero objeto (obtenido con gcc -c)**

a) no reserva espacio para las direcciones de objetos desconocidos; p.ej., sólo al enlazar un CALL a printf se insertan los 4B de la dirección de printf entre el codop de CALL y el de la siguiente instrucción máquina

el espacio reservado para símbolos reubicables/indefinidos aparece relleno con 0

b) contiene tablas de símbolos, pero sólo para los símbolos locales; p.ej., sólo al enlazar un programa que llame a printf se añadirá una entrada a la tabla con la dirección de printf

los símbolos no resueltos/no definidos aparecen etiquetados U / \*UND\*

-> c) ubica el código (y los datos) a partir de la posición 0x0, las direcciones definitivas sólo se calculan tras enlazar

d) no contiene tablas de símbolos, que sólo se añaden al ejecutable tras enlazar

objdump -t muestra la tabla de símbolos, de ejecutables y/o objetos

**8. En X86-64, el registro contador de programa se denomina:**

- a) R15
- b) IP
- c) EIP
- > d) RIP

**9. ¿Cuántos operandos acompañan a la instrucción PUSH en IA32?**

- a) 0
- > b) 1
- c) 2
- d) 3

**10. Dada la siguiente definición de datos:**

lista: .int 0x10000000, 0x50000000,

          0x10000000, 0x20000000

longlista: .int (. - lista)/4

resultado: .quad 0x123456789ABCDEF

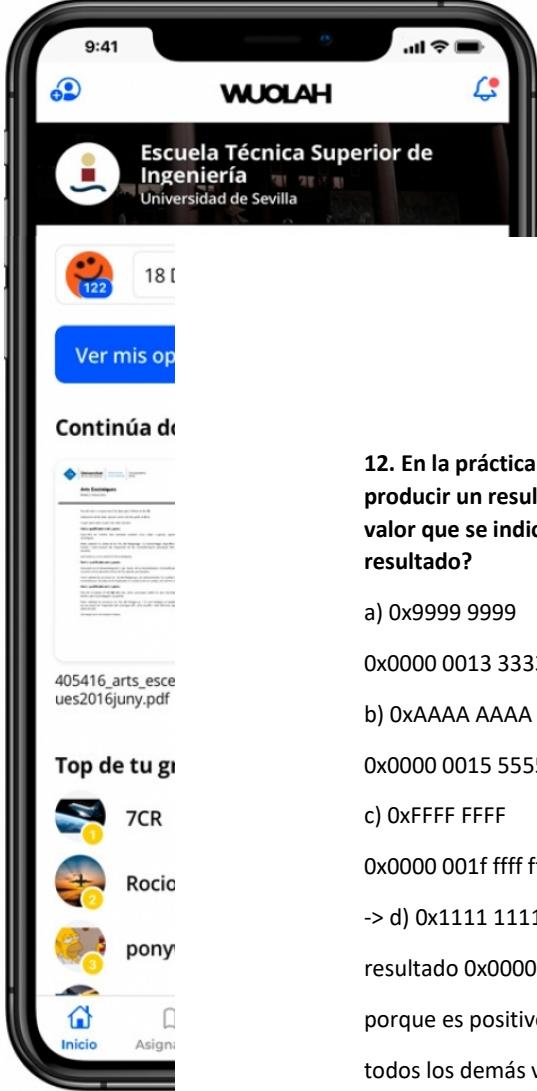
formato: .ascii "suma=%llu=%llx hex\n\0"

**la instrucción movl longlista, %ecx copia el siguiente valor:**

- a) 8
- > b) 4
- c) 16
- d) 32

**11. ¿Qué modificador (switch) de as hace falta para ensamblar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?**

- a) -m32
- > b) -32
- c) -m elf\_i386
- d) Ninguna de las anteriores respuestas es correcta



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**12.** En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor que se indica a continuación, ¿en qué caso ambos programas producen el mismo resultado?

a) 0x9999 9999

0x0000 0013 3333 3320 != 0xffff fff3 3333 3320

b) 0xAAAA AAAA

0x0000 0015 5555 5540 != 0xffff fff5 5555 5540

c) 0xFFFF FFFF

0x0000 001f ffff ffe0 != 0xffff ffff ffff ffe0

-> d) 0x1111 1111

resultado 0x0000 0002 2222 2220

porque es positivo incluso en complemento a 2

todos los demás valores se interpretan como negativos, lo primero que hace la suma con signo es extenderlos a 64bit de manera que se activan los 32 bits superiores... resultado radicalmente distinto

Recordar que multiplicar por 32 es desplazar 5 posiciones a la izquierda

**13. Compilar .s → ejecutable, usando sólo as y ld, sin gcc...**

a) No se puede

-> b) Se puede, usando en as y ld los modificadores (switches) que corresponda

c) Basta usar as, con los modificadores que corresponda

d) Basta usar ld, con los modificadores que corresponda

**14. Si ECX vale 0, la instrucción adc \$-1,%ecx**

Usuario Profesores

-> a) No cambia CF

si CF=1 al sumar 0xff...ff(=4G-1) sale CF=1, si no, no

b) Cambia CF

c) Pone CF=0

d) Pone CF=1

**15. ¿Cuál de los siguientes grupos de instrucciones IA32 sólo modifican los indicadores de estado sin almacenar el resultado de la operación?**

- a) IMUL, IDIV
- > b) CMP, TEST
- c) ADC, SBB
- d) AND, OR, XOR

**16. Los switches --32 y --64 para trabajar en 32bit/64bit corresponden a la herramienta...**

- a) ld

sería -melf\_i386 y -melf\_x86\_64

- > b) as

- c) gcc

sería -m32 y -m64

- d) nm

no tiene 32/64 bits

**17. ¿Qué modificador (switch) de gcc hace falta para compilar .c→.s (de fuente C a fuente ASM)?**

- > a) gcc -S

- b) gcc -s

- c) gcc -c

- d) Eso no se puede hacer con gcc

**18. En x86-64 se pueden referenciar los registros**

- a) %r12q, %r12d, %r12w, %r12l

sería %r12 y %r12b, no %r12q ni %r12l

- b) %r8, %r8d, %r8w, %r8l

sería %r8b, no %r8l

- c) %rsi, %esi, %si, %sih, %sil

no existe %sih

- > d) %rax, %eax, %ax, %ah, %al

Ver libro Hallaron Figura 3.35.

%ah no sería compatible con los nuevos nombres x86-64 como %sib, %r8b.

**19. En IA32, tras dividir 0x640000 (64 bits) entre 0x8000 (32 bits), el resultado será:**

- a) 0x0 en EAX y 0xC8 en EDX

mal: cociente 0 (saber que cociente es en EAX)

- b) 0x0 en AX y 0xC8 en DX

mal: AX/DX

- c) 0xC8 en AX y 0xC8 en DX

mal: resto no cero (0x640 es múltiplo de 0x8)

- > d) 0xC8 en EAX y 0x0 en EDX

por exclusión, y además  $0x640 \gg 3 = 0xC8$

**20. En la práctica "media" se pide calcular la media y resto de una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Qué (media : resto) se debe obtener para una lista rellena a -1 salvo el primer elemento, que valiera -31?**

- a) (-2 : 2)

- b) (-2 : 1)

- c) (-1 :-31)

- > d) (-1 :-30)

**22. En la práctica "media" se desea invocar desde lenguaje ensamblador la función printf() de libC. Eso implica que este programa, como todo programa que use libC,**

- a) es ventajoso para ensamblarlo que contenga main, y entonces es conveniente enlazarlo usando gcc (aunque ambas cosas son opcionales)

No. Que exista main o \_start sólo se comprueba a la hora de linkar. Y lo apropiado es que un programa ensamblador empiece por \_start. Es el runtime de C quien define un \_start que invoca a main.

- > b) es ventajoso para enlazarlo usar gcc, y entonces es conveniente que contenga main (aunque ambas cosas son opcionales)

Sí. Ver comentarios para formato, main y call printf en P2 Figura 12 y P3 Figura 4

- c) es obligatorio que contenga main (y entonces es más cómodo usar gcc para enlazar)

No. Se puede usar ld -lc -dynamic-linker, sin incluir runtime

- d) es obligatorio enlazarlo usando gcc (y entonces es más cómodo que contenga main)

**21. El volcado mostrado abajo se ha obtenido con el comando...**

**00000000 <main>:**

```
0: 55 push %ebp
1: 89 e5 mov %esp,%ebp
3: 83 e4 f0 and $-16,%esp
6: 83 ec 10 sub $0x10,%esp
9: c7 44 24 movl $3, 4(%esp)
```

**c: 04 03 00 00 00**

**11: c7 04 24 movl \$0x1,(%esp)**

**14: 01 00 00 00**

**18: e8 fc ff ff ff call <main+0x19>**

**1d: c9 leave**

**1e: c3 ret**

a) objdump -t p2.o

-> b) objdump -d p2.o

-d porque es un desensamblado (no cabeceras -h ni tablas -t)

p2.o porque las direcciones empiezan en 0 (un ejecutable x86 empezaría en 0x080480xx)

c) objdump -d p

d) objdump -h p

**23. ¿Qué modificador (switch) de gcc hace falta para compilar .s → .o sin llamar al enlazador?**

a) Eso no se puede hacer con gcc

-> b) gcc -c

c) gcc -s

d) gcc -S

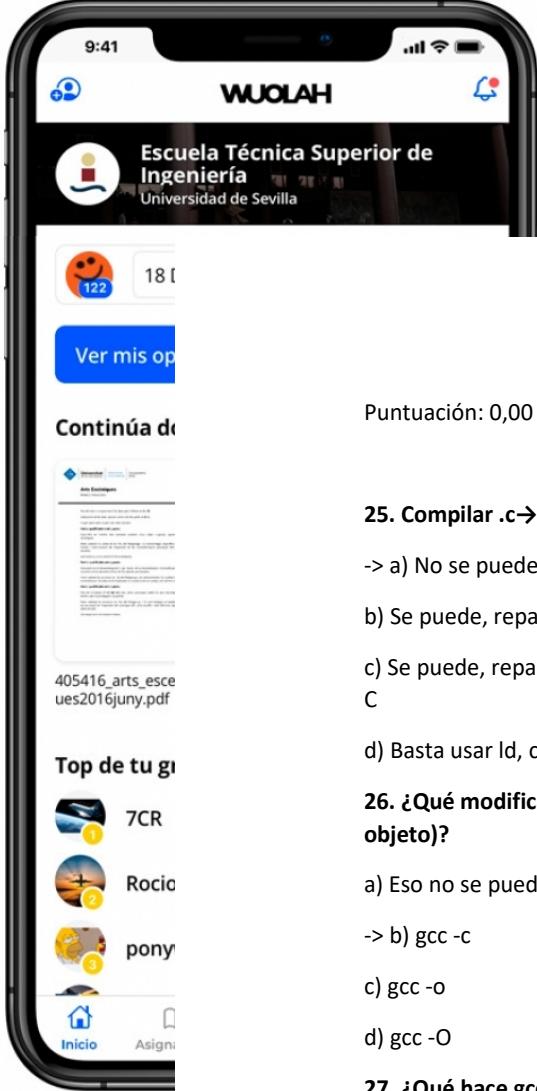
**24.. ¿Qué hace gcc -O?**

-> a) Compilar con optimización suave

b) Compilar .c→.o (fuente C a objeto)

c) Compilar .s→.o (fuente ASM a objeto)

d) Ambas (b) y (c), según la extensión de los ficheros que se usen como argumentos



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## 25. Compilar .c→exe (de fuente C a ejecutable) usando sólo as y ld, sin gcc...

- > a) No se puede
- b) Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
- c) Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C
- d) Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C

## 26. ¿Qué modificador (switch) de gcc hace falta para compilar .c → .o (de fuente C a código objeto)?

- a) Eso no se puede hacer con gcc
- > b) gcc -c
- c) gcc -o
- d) gcc -O

## 27. ¿Qué hace gcc -O1?

- a) Compilar .c→.o (fuente C a objeto)
- b) Compilar .s→.o (fuente ASM a objeto)
- c) Ambas (a) y (b), según la extensión de los ficheros que se usen como argumentos
- > d) Compilar con optimización

## 28. En la práctica “media” se pide sumar una lista de 32 enteros \*con\* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en toda la lista causaría overflow con 32bits?

- a) 0x8000 0000

ya es el negativo más grande, en cuanto se le sumara -1 saldría overflow

- b) 0x4000 0000

en cuanto se hiciera la primera suma saldría negativo

- > c) 0x0400 0000

$n*32 == n << 5$  y el bit 26 (0x04...) pasaría a 31 (0x80...) y el resultado llegaría \*justo\* a convertirse en negativo. Algo más pequeño que 0x04... no llegaría.

- d) 0x0800 0000

se perdería incluso el bit de signo

**29. ¿Qué hace gcc -O0?**

- a) Compilar .c → .s (C → ASM sin generar objeto)
- b) Compilar .s → .o (fuente ASM a objeto)
- c) Compilar .c → .o (fuente C a objeto)
- > d) Compilar sin optimización

**30. Tras ejecutar las tres instrucciones que se muestran desensambladas a continuación, el registro EAX toma el valor**

**08048074 <\_start>:**

**8048074: be 74 80 04 08 mov \$\_start, %esi**

**8048079: 46 inc %esi**

**804807a: 8b 06 mov (%esi), %eax**

- > a) 0x08048074
- b) 0x08048075
- c) 0x08048079
- d) 0x0804807a

**31. ¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?**

- a) -64
- b) -32
- > c) -m32
- d) -m64

**32. En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando perder acarreos. ¿Cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32bits (sin signo)?**

- a) 0xfc00 0000
- b) 0xfbff ffff
- c) 0x07ff ffff
- > d) 0x0800 0000

**33. Como parte del proceso de compilación de una aplicación en lenguaje C, enlazar .o → .exe (de objeto proveniente de fuente C a ejecutable) usando sólo as y ld, sin gcc...**

- a) Ninguna de las anteriores respuestas es correcta
- > b) Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C
- c) Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
- d) Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C

**34. Los switches –m elf\_i386 y –m elf\_x86\_64 para trabajar en 32 bits / 64 bits corresponden a la herramienta...**

- a) nm
- > b) ld
- c) gcc
- d) as

**35. ¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa en lenguaje C? (el fichero sin extensión es un ejecutable):**

- a) file.s → file.c → file → file.o
- > b) file.c → file.s → file.o → file
- c) file.s → file.c → file.o → file
- d) file.c → file.s → file → file.o

**36. ¿Cuál de los siguientes contenidos no está incluido en un fichero ELF ejecutable?**

- a) código máquina
- sección .text
- b) variables globales
- secciones .data/.bss
- c) tabla de símbolos
- incluso símbolos de depuración con –g
- > d) pila del usuario

**37. La siguiente línea en la sección de datos de un programa en ensamblador de IA32**

**result: .int 0,0**

- a) Reserva espacio para un único entero, inicializado a 0, en la posición de memoria 0
- b) Reserva espacio para un único entero, inicializado a 0,0
- > c) Reserva espacio para dos enteros, inicializados ambos a 0
- d) Reserva espacio para un entero, inicializado a 0, seguido de un dato de tamaño indefinido, también inicializado a 0

**38. Considerar los siguientes dos bloques de código almacenados en dos ficheros distintos:**

```
/* main.c */
```

```
int i = 0;
```

```
int main() {
```

```
 func();
```

```
 return 0;
```

```
}
```

```
/* func.c */
```

```
int i = 1;
```

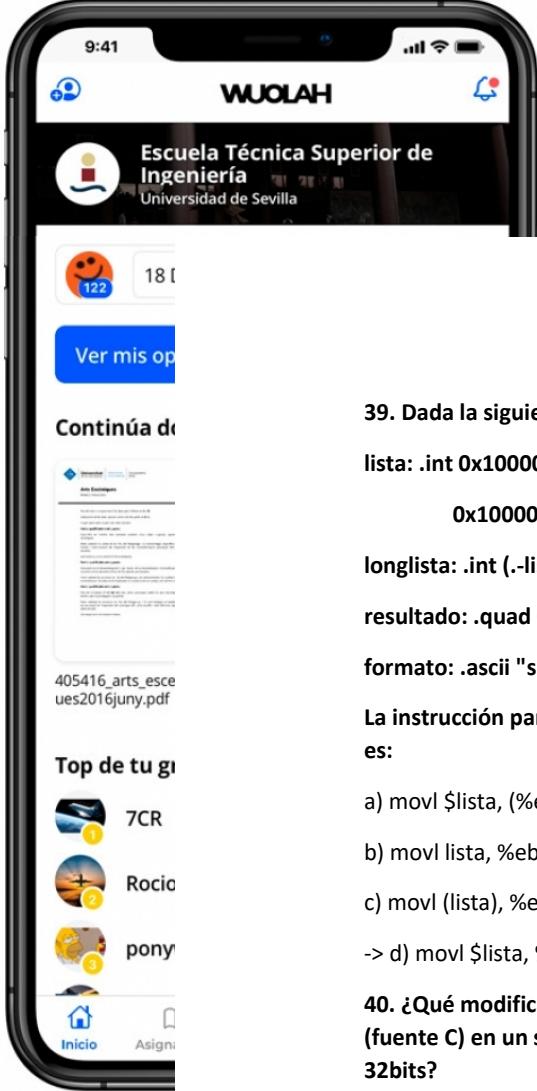
```
void func() {
```

```
 printf("%d", i);
```

```
}
```

**¿Qué sucederá cuando se compile, enlace y ejecute este código?**

- > a) Error al compilar o enlazar, no se obtiene ejecutable
- b) Escribe “0”
- c) A veces escribe “0” y a veces “1”
- d) Escribe “1”



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**39. Dada la siguiente definición de datos:**

lista: .int 0x10000000, 0x50000000,

0x10000000, 0x20000000

longlista: .int (-lista)/4

resultado: .quad 0x123456789ABCDEF

formato: .ascii "suma=%llu=%llx hex\n\0"

**La instrucción para copiar la dirección de memoria donde comienza lista en el registro EBX es:**

a) movl \$lista, (%ebx)

b) movl lista, %ebx

c) movl (lista), %ebx

-> d) movl \$lista, %ebx

**40. ¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32bits (fuente C) en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?**

a) No se puede conseguir ese efecto con modificadores de gcc, porque el gcc por defecto (/usr/bin/gcc) es el de 64bits, se debe usar el gcc de la instalación alternativa (de 32bits)

b) -m elf\_i386

-> c) -m32

d) -32

**41. En la práctica “media” se pide sumar una lista de enteros \*con\* signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en los \*dos\* primeros elementos de la lista causaría overflow con 32 bits al realizar la suma de \*esos dos\* primeros elementos de la lista?**

a) 0x8000 0000

es negativo, el enunciado pide menor valor positivo

-> b) 0x4000 0000

0x8000 0000 sería negativo en 32bit

c) 0x0400 0000

suma 0x0800 0000 sin problema

d) 0x0800 0000

suma 0x1000 0000 sin problema

**42. En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. Un estudiante entrega la siguiente versión**

# \$lista en EBX, longlista en ECX

suma:

```
mov $0, %eax
mov $0, %edx
mov $0, %esi
```

bucle:

```
add (%ebx,%edx,4), %eax
jnc seguir
inc %edx
```

seguir:

```
inc %esi
cmp %esi,%ecx
jne bucle
ret
```

**Esta función presenta una única diferencia frente a la solución recomendada en clase, relativa al indexado en el array.**

**Esta función suma:**

- a) Produce siempre el resultado correcto
- b) Fallaría con lista: .int 1,1,1,1, 1,1,1,1, ...

Coincide que sumar 32 veces lista[0]=1 es lo mismo que sumar los 32 elementos, porque todos son iguales a lista[0]

-> c) Fallaría con lista: .int 1,2,3,4, 1,2,3,4, ...

Debería salir suma=80 y sale suma=32 porque suma 32 veces lista[0]=1.  $32 <<< 4G - 1$ , así que no hay acarreos ni por asomo.

- d) No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**43. Dada la siguiente definición de datos:**

lista: .int 0x10000000, 0x50000000,

          0x10000000, 0x20000000

longlista: .int (-.lista)/4

resultado: .quad 0x123456789ABCDEF

formato: .ascii "suma=%lu=%lx hex\n\0"

la llamada correcta a printf será:

a) push resultado

    push resultado+4

    push resultado

    push resultado+4

    push \$formato

    call printf

    add \$20, %esp

b) push resultado

    push resultado+4

    push \$formato

    call printf

    add \$12, %esp

c) push resultado+4

    push resultado

    push \$formato

    call printf

    add \$12, %esp

-> d) push resultado+4

    push resultado

    push resultado+4

    push resultado

    push \$formato

    call printf

    add \$20, %esp

**44. ¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa en lenguaje C? (el fichero sin extensión es un ejecutable):**

- a) fich.c → fich.o → fich.s → fich
- b) fich → fich.s → fich.o → fich.c
- > c) fich.c → fich.s → fich.o → fich
- d) fich.c → fich.s → fich → fich.o

**45. ¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?**

- a) No hace falta modificador, ld lo deduce del tipo de objeto a enlazar
- > b) -m elf\_i386
- c) -m32
- d) -32

**46. En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando desbordamiento. De entre los siguientes, ¿cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32**

**bits (en complemento a 2)? Se usa notación decimal y espacios como separadores de millares/millones/etc.**

- a) -10 000 000

No llega, abs(-10 millones) <= 64M

- > b) -100 000 000

Se pasa, 100 millones >> 64M

- c) -1 000 000 000

- d) -10 000 000 000

**47. En la práctica “media” se pide sumar una lista de 32 enteros \*con\* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista causaría overflow con 32bits?**

- a) 0xffff ffff

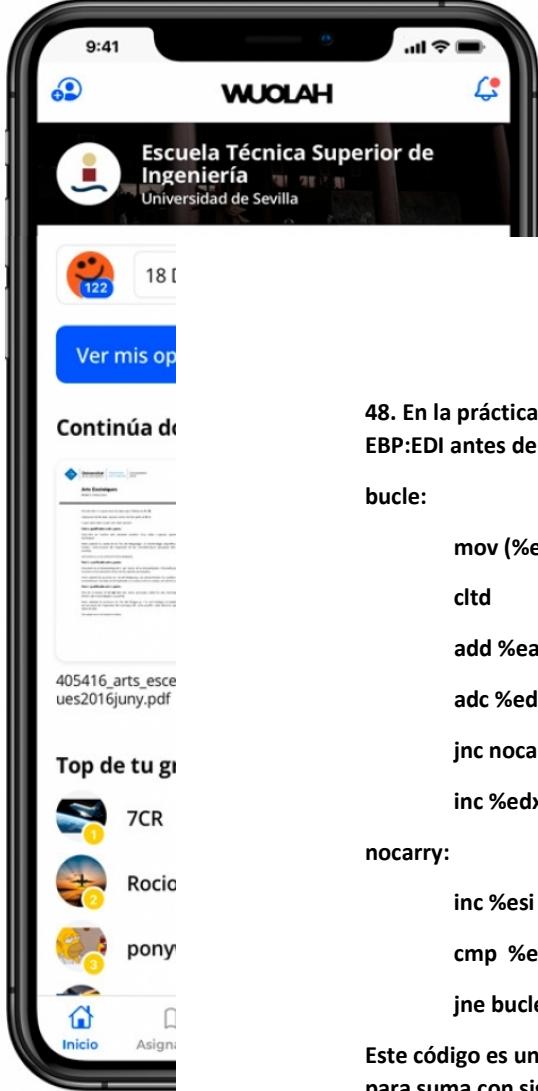
- b) 0xfc00 0000

nx32 == n<<5 y quedaría 0x8000 0000 !!! justo para que no haya overflow

- c) 0xf000 0000

se pierde el signo con <<4, mucho más con <<5

- > d) 0xfbff ffff



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

48. En la práctica "media" un estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

bucle:

```
mov (%ebx,%esi,4), %eax
```

```
cltd
```

```
add %eax, %edi
```

```
adc %edx, %ebp
```

```
jnc nocarry
```

```
inc %edx
```

nocarry:

```
inc %esi
```

```
cmp %esi,%ecx
```

```
jne bucle
```

Este código es una mezcla inexplicada de las soluciones recomendadas para suma sin signo y para suma con signo. Estando bien programado todo lo demás, este código

-> a) produce siempre el resultado correcto

inc %edx se hace después de adc, cuando ya se ha usado el valor de %edx

Recordar que usamos cltd para extender a %edx el elemento leido en %eax

Igual podríamos haber hecho xor %edx,%edx y tampoco pasaría nada

b) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

c) fallaría con lista: .int 0,1,2,3

d) fallaría con lista: .int -1,-2,-4,-8

49. ¿Qué hace gcc -O?

a) Compilar sin optimización, igual que -O0

b) Compilar .c → .o

-> c) Compilar con optimización, igual que -O1

d) Compilar .s → .o

**50. En la práctica "media" se suma una lista de 32 enteros de 4 B con signo para producir una media y un resto usando la instrucción IDIV. ¿Cuál de las siguientes afirmaciones es falsa?**

- a) IDIV produce el mismo resto que el operador % en lenguaje C  
división truncada ambos - resto del mismo signo que dividendo
- b) IDIV produce el mismo cociente que el operador / en lenguaje C  
división truncada ambos - resto del mismo signo que dividendo
- c) El resto siempre tiene el mismo signo que la suma  
división truncada IDIV - resto del mismo signo que dividendo
- > d) La media se redondea al entero más próximo  
división truncada IDIV - no se redondea, se trunca

**51. Para compilar un programa escrito en C en el entorno GNU/Linux se usa el programa:**

- a) gdb
- b) objdump
- > c) gcc
- d)ddd

**52. La(s) instrucción(es) necesaria(s) para cargar el dividendo 0xa30bf18a en la pareja edx:eax como paso previo a una división sin signo son:**

- a) movq \$0xa30bf18a,%rax
- > b) movl \$0xa30bf18a,%eax  
xorl %edx,%edx
- c) movl \$0xf18a,%eax  
movl \$0xa30b,%edx
- d) movl \$0xa30bf18a,%eax  
cltd

**53. En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32bits (en complemento a 2)?**

- a) 0xf800 0000
- > b) 0xfbff ffff
- c) 0xfc00 0000
- d) 0xf800 0001

**54. Dada la siguiente definición de datos:**

lista: .int 0x10000000, 0x50000000,

          0x10000000, 0x20000000

longlista: .int (-.lista)/4

resultado: .quad 0x123456789ABCDEF

formato: .ascii "suma=%llu=%llx hex\n\0"

y suponiendo que hemos llamado a una función suma que devuelve un número de 64 bits en la pareja EDX:EAX, las instrucciones que copian ese número en resultado son:

a) movl (%eax), resultado

      movl (%edx), resultado+4

      mov M-M imposible

b) movl %eax, resultado+4

      movl %edx, resultado

      no es little-endian (tampoco es big-)

c) movl (%eax), resultado+4

      movl (%edx), resultado

      mov M-M imposible

-> d) movl %eax, resultado

      movl %edx, resultado+4

      little-endian => primero el menos significativo

**55. El switch de gcc para que únicamente compile de lenguaje C a ensamblador, y no realice ningún paso adicional (ensamblar, enlazar, etc), es...**

a) -o

      para nombrar el ejecutable

-> b) -S

c) -c

      de C/asm a objeto, .c/.s→.o

d) -g

      para incorporar info depuración

**56. El switch -l para indicar librerías \*NO\* funciona con la herramienta...**

a) gcc

gcc pasa -L/-l a ld

-> b) as

c) ld

-L/-l son switches propios de ld

d) no se puede marcar una y solo una de las anteriores

**57. En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x0400 0000, ¿en qué se diferencian los resultados de ambos programas?**

a) en uno ocupa 32 bits, en otro 64 bits

ambos resultados son de 64 bits

-> b) no se diferencian

0x0000 0000 8000 0000

el resultado cabría en 32bits sin signo

c) en uno se interpreta como negativo, en otro como positivo

ambos programas interpretan 0x0000 0000 8000 0000 como positivo

d) en uno los 32 bits superiores son 0xFFFF FFFF, en el otro no

en ambos los 32bits superiores son 0x0000 0000

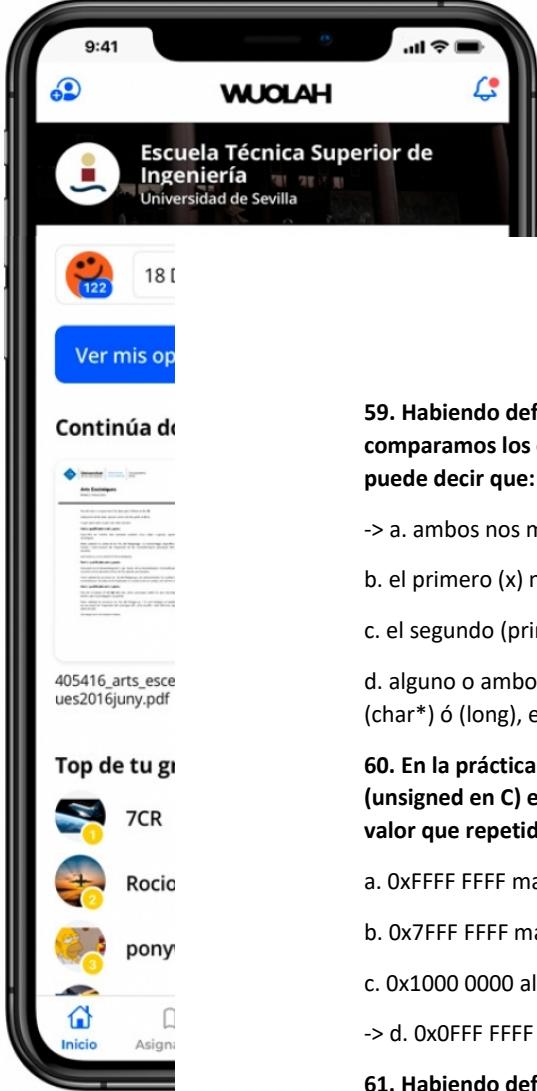
**58. Habiendo definido en código fuente ensamblador longsal: .quad .-saludo justo detrás de un string saludo que ocupaba 28 bytes, si comparamos los comandos gdb siguientes: x/1xg &longsal frente a print (long) &longsal, se puede decir que:**

a. ambos nos muestran la longitud del string (que es/vale/equivale a 28)

-> b. el primero (x) nos muestra la longitud, y el segundo (print) nos muestra otro valor distinto  
Práct.2, pág.7, Fig.11

c. el segundo (print) nos muestra la longitud, y el primero (x) nos muestra otro valor distinto

d. alguno o ambos contienen algún error gramatical (falta o sobra algún &, algún typecast (char\*) ó (long), etc)



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**59. Habiendo definido en código fuente ensamblador saludo: .ascii "Hola mundo\n"), si comparamos los comandos gdb siguientes: x/s &saludo frente a print (char\*) &saludo, se puede decir que:**

- > a. ambos nos muestran el string (el mensaje "Hola mundo") Práct.2, pág.7, Fig.11
- b. el primero (x) nos muestra el string, y el segundo (print) nos muestra otro valor distinto
- c. el segundo (print) nos muestra el string, y el primero (x) nos muestra otro valor distinto
- d. alguno o ambos contienen algún error gramatical (falta o sobra algún &, algún typecast (char\*) ó (long), etc)

**60. En la práctica "media" se pide sumar una lista de 16 enteros \*sin\* signo de 32 bits (unsigned en C) evitando desbordamiento (sin signo), esto es, acarreo. ¿Cuál es el mayor valor que repetido en toda la lista \*no\* causaría acarreo en 32 bits (unsigned en C)?**

- a. 0xFFFF FFFF mayor que b.
- b. 0x7FFF FFFF mayor que c.
- c. 0x1000 0000 al desplazar 4 bits a la izquierda sale el 1 hacia el CF
- > d. 0x0FFF FFFF

**61. Habiendo definido en código fuente ensamblador saludo: .ascii "Hola a todos!\nHello World!\n"), para conseguir desde gdb cambiar el string para que la llamada al sistema WRITE escriba solo una línea en lugar de dos, se debería hacer...**

- a. nada, porque ya de por sí WRITE escribe saludo como una sola línea
- b. nada, desde gdb no se puede conseguir que WRITE escribe saludo como una sola línea
- c. p (char\*) &saludo+13 y también p (char\*) &saludo+14\_
- > d. set var \*((char\*)&saludo+13)='-'

**62. En la práctica "media" se pide sumar una lista de 16 enteros \*sin\* signo de 32 bits (unsigned en C) evitando desbordamiento (sin signo), esto es, acarreo. ¿Cuál es el menor valor que repetido en toda la lista causaría acarreo en 32 bits (unsigned en C)?**

- a. 0xFFFF FFFF
- b. 0x7FFF FFFF
- > c. 0x1000 0000
- //al desplazar 4 bits a la izquierda sale el 1 hacia el CF
- d. 0x0FFF FFFF

**63. En un sistema Linux x86-64, ¿cuál de las siguientes expresiones es equivalente a la expresión C  $(x[2] + 4)[3]$ ? Suponer que previamente se ha declarado int \*\*x. x[2] es int\*, y entonces da igual  $(X[2])[7] == (X[2]+7) == (X[2]+4)[3]$**

- a.  $*((*(x + 16)) + 28)$

cierto si se hubiese hecho typecast (void\*)x para evitar aritmética punteros

- b.  $*(((*x) + 2) + 7)$

$(*x)+2 == x[0]+2 != x[2]$

- c.  $(*(*x + 2) + 4) + 3$

sería  $x[2][4]+3$ , la última suma sería suma entera, no aritmética punteros

- d.  $*((*(x + 2) + 4) + 3)$

# TEST EC PRACTICA 3

1. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity3:

```
int parity3(unsigned* array, int len){
 int i,res=0,val;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
 do {
 val += x;
 x >>= 1;
 } while (x);
 val &= 0x1;
 res+=val;
 }
 return res;
}
```

Esta función parity3:

-> a) produce siempre el resultado correcto

solemos escribir `res+=val&0x1`, en lugar de ponerlo en 2 sentencias C

b) fallaría con `array={0,1,2,3}`

c) fallaría con `array={1,2,4,8}`

d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

2. ¿Cuál es el **popcount** (peso Hamming, nº de bits activados) de una lista de N números inicializada con los valores 0..N-1?

a)  $(N-1)*N/2$

no, eso es la fórmula de Gauss para la suma de la progresión aritmética

b)  $N*(N+1)/2$

no, eso sería la suma de la progresión aritmética de 1..N

c) si N es par,  $N*(N/2)$

no, en todo caso  $N/2$  sería la suma de paridades

-> d) si N es potencia de 2,  $(\log_2(N)*N)/2$

todas las combinaciones de  $\log_2(N)$  bits, en cada posición de bit la mitad son 0 y la otra mitad son 1

p.ej.: LSB mitad par mitad impar, MSB mitad empieza por 0 mitad por 1, etc...

### 3. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%[r]"
```

```
 :[r] "+r" (result)
```

```
 :[i] "r" (i),
```

```
 [a] "r" (array)
```

```
);
```

¿Cuál de las siguientes afirmaciones es incorrecta?

a) se desea que el valor calculado por la instrucción ensamblador quede almacenado en la variable C result

correcto, add ...,%[r] almacena en result

b) r es un registro de entrada/salida

ok, "+r"

c) a es un registro de entrada

ok, "r"

-> d) i es un registro de salida

no, asm("sentence":output:input:clobber), [i] va en input

### 4. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount4:

```
int popcount4(unsigned* array,
```

```
 int len){
```

```
 int i,j;
```

```

unsigned x;
int result = 0;
for(i=0;i<len;i++){
 x=array[i];
 for(j=0;j<8;j++){
 result += x & 0x01010101;
 x>>=1;
 }
 result += (result >> 16);
 result += (result >> 8);
 return result & 0xFF;
}

```

**Esta función presenta varias diferencias con la versión "oficial" recomendada en clase, incluyendo la ausencia de una variable auxiliar val y la diferente posición de los desplazamientos >> y máscara 0xFF.**

**Esta función `popcount4`:**

a) produce siempre el resultado correcto

No. Las sumas paralelas pueden llevar acarreos al siguiente byte. Y la suma total puede ser superior a 255, cosa que no permite el código mostrado.

b) fallaría con `array={0,1,2,3}`

No. La suma total es 4, no hay acarreo al byte superior ( $4 << 256$ ), quedan sin efecto las instrucciones erróneas o cambiadas de sitio ( $>>16$ ,  $>>8$ ,  $\&0xff$ )

c) fallaría con `array={1,2,4,8}`

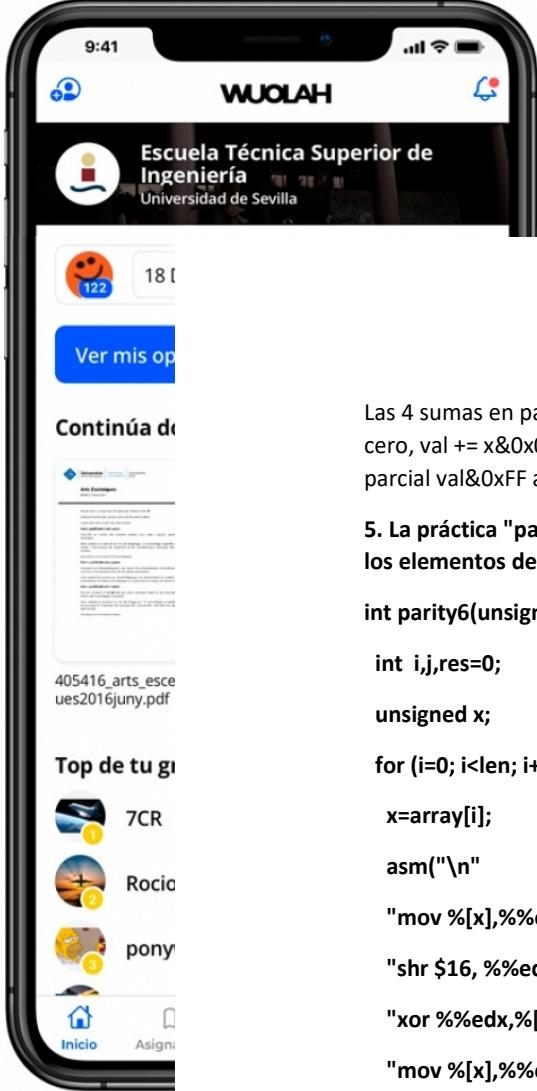
No. La suma total es 4, no hay acarreo al byte superior ( $4 << 256$ ), quedan sin efecto las instrucciones erróneas o cambiadas de sitio ( $>>16$ ,  $>>8$ ,  $\&0xff$ )

-> d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Efectivamente.

El ejemplo más pequeño para llegar al error (suma paralela menos significativa  $>= 256$ ) sería  
`lista[32]={0xff, 0xff, ... 0xff}`

En ese caso, calcula `result=256`, le suma 1 (`result=257`) y lo enmascara: `result=1`



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

[Available on the App Store](#) [GET IT ON Google Play](#)

Las 4 sumas en paralelo de  $x \& 0x01010101$  deben hacerse sobre una variable inicialmente a cero,  $val += x \& 0x01010101$ , y luego acumularse  $val += val >> 16$  y 8, antes de acumular la suma parcial  $val \& 0xFF$  a result.

5. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned* array, int len){
 int i,j,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 asm("\n"
 "mov %[x],%%edx \n\t"
 "shr $16, %%edx \n\t"
 "xor %%edx,%[x] \n\t"
 "mov %[x],%%edx \n\t"
 "mov %%dh, %%dl \n\t"
 "xor %%edx, %[x]\n\t"
 "setpo %%cl \n\t"
 "movzx %%cl, %[x]"
 "[x] "+r" (x)
 :
 :"edx","ecx"
);
 res+=x;
 }
 return res;
}
```

Esta función parity6:

-> a) produce siempre el resultado correcto

bastaba con un sobrescrito "edx", y con 6 instrucciones máquina (no 8)

en concreto, la 3<sup>a</sup> instrucción podría haber sido xor %[x],%%edx y ya sólo quedaría operar con %dh y %dl, y no usar %cl sino el propio %dl

- b) fallaría con array={0,1,2,3}
- c) fallaría con array={1,2,4,8}
- d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**6. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de `popcount4`:**

```
int popcount4(unsigned* array, int len)
```

```
{
 int i, j, res = 0;
 for(i = 0; i < len; ++i) {
 unsigned x = array[i];
 int n = 0;
 do {
 n += x & 0x01010101L;
 x >>= 1;
 } while(x);
 for(j = 16; j == 1; j /= 2){
 n ^= (n >>= j);
 }
 res += n & 0xff;
 }
 return res;
}
```

**Esta función `popcount4`:**

-> a) produce el resultado correcto

Caso real, entregado en prácticas. La máscara está pensada para `for(j=0;j<8;j++)`. En lugar de eso, se hace `do...while(x)`, con lo cual no se ahoran iteraciones y todo el resultado queda acumulado en el LSB de `n`. El `for(j)` es absurdo, no itera ninguna vez. El resultado se extrae y acumula con `n&0xFF`. Es correcto, pero no mejora la eficiencia. `popcount2` es igual de eficiente y más elegante, porque no tiene código superfluo.

- b) fallaría con array={0,1,2,3}
- c) fallaría con array={1,2,4,8}
- d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**7. ¿Cuál es la paridad (XOR "lateral" de todos los bits) del número 29?**

-> a) 0

29 = 0x1D = 16+8+4+1 = 0b0001 1101 -> par -> xor sale 0

b) 1

c) 2

d) 4

**8. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity4:**

```
int parity4(unsigned* array, int len){
 int val,i,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
 asm("\n"
 "ini3: \n\t"
 "xor %[x],%[v] \n\t"
 "shr %[x] \n\t"
 "test %[x], %[x]\n\t"
 "jne ini3 \n\t"
 "[v]"+"r" (val)
 "[x]" "r" (x)
);
 val = val & 0x1;
 res+=val;
 }
 return res;
}
```

**La sentencia `asm()` del listado anterior tiene las siguientes restricciones**

a) ninguna

b) arquitectura de 32 bits

-> c) dos entradas y una salida

[v] cuenta como salida y entrada, [x] como entrada

d) un registro y dos sobreescritos (clobber)

**9. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity5:**

```
int paridad5(unsigned* array, int len) {
 int i, k, result = 0;
 unsigned x;
 for (i = 0; i < len; i++) {
 x = array[i];
 for (k = 16; k == 1; k /= 2)
 x ^= x >> k;
 result += (x & 0x01);
 }
 return result;
}
```

**Esta función paridad5:**

a) es correcta

b) falla para array={0,1,2,3}

-> c) falla para array={1,2,3,4}

d) no se puede marcar una y sólo una de las opciones anteriores

**10. En la convención cdecl estándar para arquitecturas x86 de 32 bits, el resultado de una función se devuelve usualmente en el registro:**

a) ESI

b) EBP

-> c) EAX

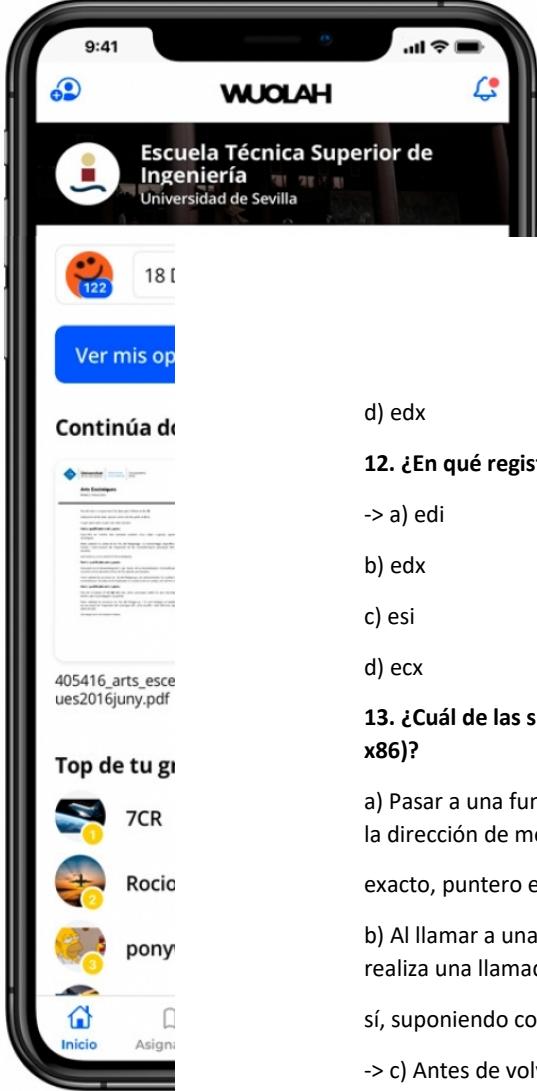
d) EBX

**11. ¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de una subrutina según la convención cdecl para IA32?**

a) eax

b) ecx

-> c) ebx



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

d) edx

**12. ¿En qué registro se pasa el primer argumento a una función en Linux gcc x86-64?**

-> a) edi

b) edx

c) esi

d) ecx

**13. ¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C (en convención cdecl x86)?**

a) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable

exacto, puntero es dirección de...

b) Al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

sí, suponiendo convención cdecl x86, porque x86-64 usa regs.

-> c) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop

no, quita parámetros el invocante, tras retornar de la llamada

d) Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero

sí, de derecha a izquierda

**14. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity5:**

```
int parity5(unsigned * array,
```

```
 int len){
```

```
 int i,j, result = 0;
```

```
 unsigned x;
```

```
 for(i = 0; i<len; i++){
```

```
 x=array[i];
```

```
 for(j=1; j<8*sizeof(int); j*=2)
```

```
 x ^= x >> j;
```

```
 result += x & 0x1;
```

```
}
```

```
 return result;
```

}

**Esta función sólo se diferencia de la versión "oficial" recomendada en clase, en las condiciones del bucle for interno.**

**Esta función parity5:**

-> a) Produce siempre el resultado correcto

Efectivamente, con este for también se cumple que el LSB (bit menos significativo) sigue siendo el XOR lateral de todos los bits.

b) Fallaría con array={0,1,2,3}

c) Fallaría con array={1,2,4,8}

d) No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

El bucle sugerido por el guión de prácticas hubiera sido:

```
for (j=8*sizeof(unsigned); j>0; j>>=1)
```

**15. Suponga la siguiente sentencia asm en un programa:**

```
asm(" add (%[a],%[i],4),%[r]"
```

```
:[r] "+r" (result)
```

```
:[i] "r" (i),
```

```
[a] "r" (array)
```

```
);
```

**¿Cuál de las siguientes afirmaciones es correcta?**

a) el código de retorno de la función asm se fuerza a que esté en la variable result

b) a es una posición de memoria de entrada

c) r es una posición de memoria de entrada/salida

-> d) i es un registro de entrada

**16. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:**

```
int popcount3(unsigned* array,
```

```
int len){
```

```
int i, res = 0;
```

```
unsigned x;
```

```
for(i = 0; i < len; i++) {
```

```

x = array[i];

asm("ini3: \n"
 "shr %[x] \n"
 "adc $0, %[r] \n"
 "add $0, %[x] \n"
 "jne ini3 \n"
 : [r] "+r" (res)
 : [x] "r" (x);
}

return res;
}

```

**Esta función sólo tiene una diferencia con la versión "oficial" recomendada en clase. En concreto, una instrucción máquina en la sección asm() es distinta.**

**Esta función popcount3:**

-> a) produce siempre el resultado correcto

Si sumar x+0 activa ZF sólo puede ser porque ya era x==0, así que la lógica es equivalente a la deseada.

b) fallaría con array={0,1,2,3}

c) fallaría con array={1,2,4,8}

d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Para acabar el bucle cuando x==0 hubiéramos esperado que se usara test %[x], %[x], que activa el flag ZF si el resultado es 0. Notar que and(x,x) sólo puede ser 0 si ya era x==0

**17. ¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?**

a) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la variable

ptr!=val

-> b) En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

suponiendo convención cdecl x86, porque x86-64 usa regs.

c) Los parámetros se introducen en la pila en el orden en el que aparecen en la llamada de C, es decir, empezando por el primero y acabando por el último

no, de derecha a izquierda

d) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop

no, quita parámetros el invocante, tras retornar de la llamada

**18. ¿Cuál es la suma de paridades (suma de los XOR "laterales" de los bits de cada número) de una lista de N números inicializada con los valores 0..N-1?**

a)  $(N-1)*N/2 //$ no, eso es la fórmula de Gauss para la suma de la progresión aritmética

b)  $N*(N+1)/2 //$ no, eso sería la suma de la progresión aritmética de 1..N

-> c) si N es par,  $N/2 //$ exacto, al cambiar el LSB cambia la paridad

d) si N es potencia de 2,  $(\log_2(N)*N)/2 //$ eso sería el popcount

**19. ¿Cuál es la paridad (XOR "lateral" de todos los bits) del número 19?**

a) 0

-> b) 1

$19 = 0x13 = 16+3 = 0b0001\ 0011 \rightarrow$  impar -> xor sale 1

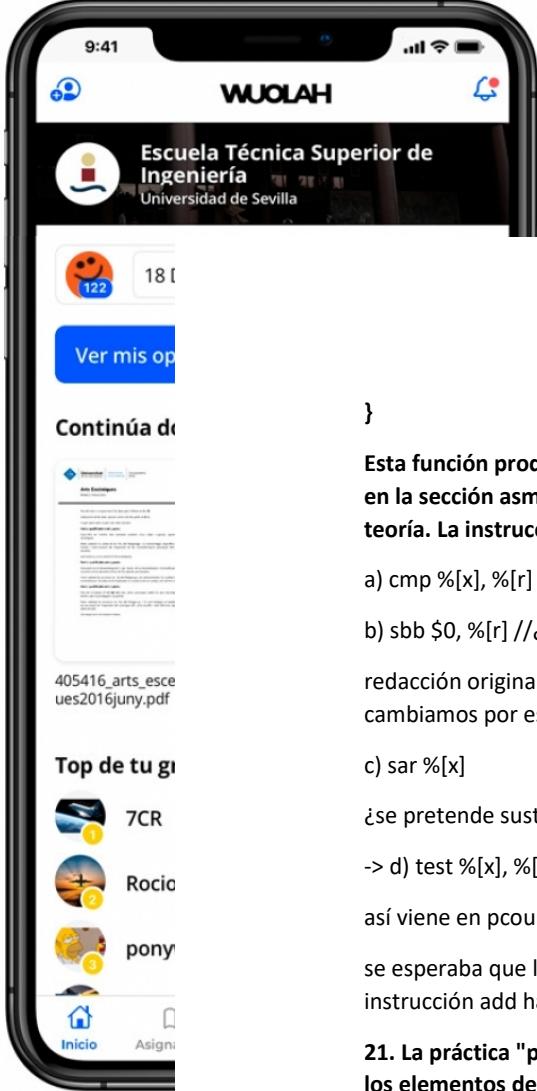
c) 2

d) 3

**20. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:**

```
int popcount3(unsigned* array,
```

```
 int len){
 int i, res = 0;
 unsigned x;
 for(i=0; i<len; i++) {
 x = array[i];
 asm("ini3: \n"
 "shr %[x] \n"
 "adc $0, %[r] \n"
 "add $0, %[x] \n"
 "jne ini3 \n"
 : [r] "+r" (res)
 : [x] "r" (x));
 }
 return res;
```



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

}

Esta función produce siempre el resultado correcto, a pesar de que una instrucción máquina en la sección asm() es distinta a la que se esperaba después de haber estudiado pcount\_r en teoría. La instrucción distinta también se podría haber cambiado por...

- a) cmp %[x], %[r] //no sirve
- b) sbb \$0, %[r] //¿se pretende sustituir la segunda instrucción?

redacción original adc \$0, %[x] hubiera servido para sustituir la 3ª instrucción, así que cambiamos por esta otra que no sirve (usa %[r]).

- c) sar %[x]
- ¿se pretende sustituir la primera instrucción?

-> d) test %[x], %[x]

así viene en pcount\_r (Tema 2.3 tr.38)

se esperaba que la 3ª instrucción, antes de jne, fuera test, para comprobar si x==0. La instrucción add hace lo mismo de una forma más difícil.

**21. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity5:**

```
int parity5(unsigned* array, int len){
 int i,j,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 for (j=sizeof(unsigned)*4;
 j>0; j=j/2){
 x^=x>>j;
 }
 x = x & 0x1;
 res+=x;
 }
 return res;
}
```

Esta función parity5:

- > a) produce siempre el resultado correcto

solemos escribir res+=x&0x1, en lugar de ponerlo en 2 sentencias C

- b) fallaría con array={0,1,2,3}
- c) fallaría con array={1,2,4,8}
- d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**22. Para averiguar la paridad de un número se puede usar la operación:**

- a) NOT
- b) OR
- c) AND
- > d) XOR

**23. Utilizando la sentencia asm(), las denominadas restricciones que se indican al final de dicha sentencia, involucran a:**

- a) Solamente las entradas
- b) Solamente las salidas
- c) Solamente los sobrescritos
- > d) Ninguna de las anteriores es cierta

**24. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:**

```
int parity6(unsigned * array, int len)
{
 int i, result = 0;
 unsigned x;
 for (i=0; i<len; i++){
 x = array[i];
 asm("mov %[x], %%edx \n\t"
 "shr $16, %%edx \n\t"
 "shr $8, %%edx \n\t"
 "xor %%edx,%%edx \n\t"
 "setp %%dl \n\t"
 "movzx %%dl,%[x] \n\t"
 : [x] "+r" (x)
 :
 : "edx"
```

```

);
 result += x;
}
return result;
}

```

**Esta función parity6:**

- a) produce el resultado correcto
  - > b) no es correcta; fallaría por ejemplo con array={0,1,2,3}
- Caso real, entregado en prácticas. Las tres primeras instrucciones asm se pierden al poner edx a 0 usando xor. Consecuentemente, se activa PF para ajustar a impar, y termina siendo x=1. Es decir, todos los elementos del array contabilizan paridad=1. El array {1,2,4,8} pasa desapercibido, pero {0,1,2,3} debería producir resultado=2<>4.
- c) no es correcta; fallaría por ejemplo con array={1,2,4,8}
  - d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**25. En la práctica “popcount/paridad”, para cronometrar sistemáticamente las diversas versiones necesitamos una función crono() a la que se le pueda pasar como argumento cuál versión queremos cronometrar. En lenguaje C esto se puede hacer con punteros a funciones. Sabiendo que todas las versiones devuelven un valor entero, el prototipo de la función crono() debería ser:**

- a) void crono( int \* func , char\* msg);  
no compila, 1er argumento es puntero a entero
- > b) void crono( int (\* func)(), char\* msg);  
ok, 1er argumento es el puntero a función devolviendo entero
- c) void crono( int \* func (), char\* msg);  
funciona pero 2warnings & 1note: tipos incompatibles int!=int\*
- d) void crono((int \*)func (), char\* msg);  
no compila, sobran paréntesis, sería prototip.func.devolv.ptr.int

**26. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:**

```

int popcount3(int* array, int len){
 long val = 0;
 int i;
 unsigned x;

```

```

for (i=0; i<len; i++){

 x= array[i];

 do{

 val += x & 0x1;

 x >>= 1;

 } while (x);

 val += (val >> 16);

 val += (val >> 8);

}

return val & 0xFF;

}

```

**Esta función es una mezcla inexplicada de las versiones "oficiales" de `popcount2` y `popcount4`, incluyendo diferencias en 2 tipos de datos, la ausencia de la variable `res` y la diferente posición de la máscara `0xFF`.**

**Esta función `popcount3`:**

a) produce siempre el resultado correcto

No. El ejemplo grande falla (`NBITS=20`), igual que fallarían otros ejemplos grandes más pequeños (desde `NBITS=7` sale que  $7 \cdot 2^6 = 448 > 255$ ).

Fallaría con cualquier ejemplo en que el resultado fuera superior a `0xFF = 255`, porque entonces `val>>8` no sería cero y se sumaría erróneamente.

b) fallaría con `array={0,1,2,3}`

No,  $4 << 255$ , siempre `val>>8 == 0`, no se suma nada erróneamente

c) fallaría con `array={1,2,4,8}`

No,  $4 << 255$ , siempre `val>>8 == 0`, no se suma nada erróneamente

-> d) no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

Efectivamente.

El ejemplo más pequeño para llegar al error (`resultado>=256`) sería

`lista[] = {0xffffffff, 0xffffffff, 0xffffffff, 0xffffffff,`

`0xffffffff, 0xffffffff, 0xffffffff, 0xffffffff}`

En ese caso, calcula `val=256`, le suma 1 (`val=257`) y lo enmascara: `val=1`

El array no es int, val no es long, queríamos haber sumado parcialmente en res el `popcount` de cada elemento... hasta el `while(x)` iba bien, sería similar a `popcount2` (`popcount1` era con `for`, `popcount2` con `while`). Sobran los desplazamientos `val+=val>>const` y la máscara `val&0xFF`.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

En Julio 2017 Problema 3 se comenta que esta función produce el resultado correcto para los tres ejemplos pequeños, y falla con el ejemplo grande ( $237 \ll 20 \cdot 2^{19} = 10M = 10\ 485\ 760$ ), sugiriendo que produce el resultado correcto siempre que queden sin efecto las "partes erróneas" del código: `val>>16, val>>8 y val&0xFF`

### 27. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%[r]"
```

```
:[r] "+r" (result)
```

```
:[i] "r" (i),
```

```
[a] "r" (array)
```

```
);
```

#### ¿Cuál de las siguientes afirmaciones es incorrecta?

- a) se desea que el valor calculado por la instrucción ensamblador quede almacenado en la variable C result

correcto, add ...,%[r] almacena en result

-> b) i es un registro de salida

no, asm("sentence":output:input:clobber), [i] va en input

c) r es un registro de entrada/salida

ok, "+r"

d) a es un registro de entrada

ok, "r"

### 28. ¿Cuál es el **popcount** (peso Hamming, nº de bits activados) del número 29?

a) 2

b) 3

-> c) 4

$29 = 0x1D = 16+8+4+1 = 0b0001\ 1101 \rightarrow \text{popcount } 4$

d) 5

### 29. En la convención cdecl estándar para arquitecturas x86 de 32 bits, cuál de las siguientes afirmaciones es cierta:

-> a) Los parámetros se pasan en pila, de derecha a izquierda; es decir, primero se pasa el último parámetro, después el penúltimo... y por fin el primero

b) Solamente es necesario salvar el registro EAX

c) Los registros EBX, ESI y EDI son salva-invocante

d) Ninguna de las anteriores es cierta

**30. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity4:**

```
int parity4(unsigned* array, int len){
 int val,i,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 val=0;
 asm("\n"
 "ini3: \n\t"
 "xor %[x],%[v] \n\t"
 "shr %[x] \n\t"
 "test %[x], %[x]\n\t"
 "jne ini3 \n\t"
 ":[v]" +r" (val)
 ":[x] "r" (x)
);
 val = val & 0x1;
 res+=val;
 }
 return res;
}
```

**Esta función parity4:**

-> a) produce siempre el resultado correcto

solemos escribir res+=val&0x1, en lugar de ponerlo en 2 sentencias C

adicionalmente, shr afecta al flag ZF de manera que sobra test %[x],%[x]

b) fallaría con array={0,1,2,3}

c) fallaría con array={1,2,4,8}

d) no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

**31. ¿Cuál es el **popcount** (peso Hamming, nº de bits activados) del número 19?**

a) 2

-> b) 3

19 = 0x13 = 16+3 = 0b0001 0011 -> popcount 3

c) 4

d) 5

**32. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:**

```
int parity6(unsigned* array, int len){
 int i,j,res=0;
 unsigned x;
 for (i=0; i<len; i++){
 x=array[i];
 asm("\n"
 "mov %[x],%%edx \n\t"
 "shr $16, %%edx \n\t"
 "xor %%edx,%[x] \n\t"
 "mov %[x],%%edx \n\t"
 "mov %%dh, %%dl \n\t"
 "xor %%edx, %[x]\n\t"
 "setpo %%cl \n\t"
 "movzx %%cl, %[x]"
 "[x] "+r" (x)
 :
 :"edx","ecx"
);
 res+=x;
 }
 return res;
}
```

**La sentencia `asm()` del listado anterior tiene las siguientes restricciones**

- a) ninguna
- b) arquitectura de 32 bits

c) dos entradas y una salida

-> d) un registro y dos sobrescritos (clobber) // el registro es [x] "+r" y sobrescritos son "edx","ecx"

**33. En 80x86, los parámetros a las subrutinas se pueden pasar:**

a) a través de variables globales

b) a través de los registros

c) a través de la pila

-> d) todas las anteriores son ciertas

**34. ¿En qué registro se pasa el primer argumento a una función según el estándar \_cdecl en una arquitectura IA32?**

a) edi

b) esi

c) eax

-> d) Las anteriores respuestas son erróneas

**35. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity4:**

```
int parity4(unsigned* array, int len){
```

```
 int i;
```

```
 unsigned x;
```

```
 int val=0, result=0;
```

```
 for (i=0; i<len; i++){
```

```
 x = array[i];
```

```
 asm("\n"
```

```
"ini: \n\t"
```

```
 "xor %[x], %[v] \n\t"
```

```
 "shr %[x] \n\t"
```

```
 "jnz ini \n\t"
```

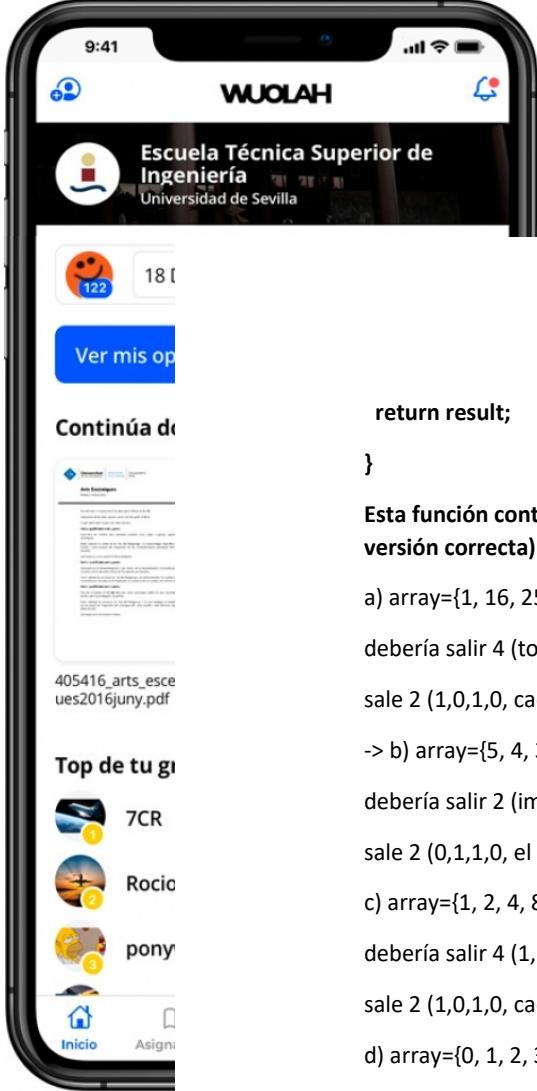
```
 : [v]"+r" (val)
```

```
 : [x] "r" (x)
```

```
);
```

```
 result += val & 0x1;
```

```
}
```



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

return result;

}

Esta función contiene un único error (en realidad se han editado 2 líneas de código sobre la versión correcta) pero produce resultado correcto cuando se usa como test el array...

a) array={1, 16, 256, 1024}

debería salir 4 (todos impares)

sale 2 (1,0,1,0, cada otro impar deshace el anterior)

-> b) array={5, 4, 3, 2}

debería salir 2 (impares el 4 y el 2)

sale 2 (0,1,1,0, el 3 cuenta como impar, el 2 deshace el impar)

c) array={1, 2, 4, 8}

debería salir 4 (1,1,1,1, todos impares)

sale 2 (1,0,1,0, cada otro impar deshace el anterior)

d) array={0, 1, 2, 3}

debería salir 2 (0,1,1,0, impares el 1 y el 2)

sale 1 (0,1,0,0, el 2 deshace el impar del 1)

El error consiste en que val=0 se inicializa al principio, en lugar de tras cada `x=array[i]`, de manera que si el LSB de val se queda activado la siguiente paridad se calcula mal. En concreto, tras calcular un impar, todos los pares que sigan cuentan como impar hasta que llegue un impar que deshaga el impar (y entonces el nuevo impar cuenta como par).

**36. Tras iniciar una sesión de depuración del Ejercicio 1 de la Práctica 3 con el comando gdb -tui --args suma\_01\_S\_SysV uno dos tres, arrancamos el programa con br\_start y run. Si tecleamos entonces p \*(char\*\*)(rsp+40), ¿qué obtenemos?**

a. tres sería \$rsp+32

-> b. 0x0 Práct.3, pág.4, Fig.3, NULL

c. una variable de entorno sería \$rsp+48, 56, 64...

d. un error de gdb

**37. ¿Cuál es el `popcount` (peso Hamming, nº de bits activados) del número 42?**

a. 2

-> b.  $3 // 42 = 0x2A = 0b10\ 1010 \Rightarrow 3 \text{ bits (3+2+2)}$

c. 4

d. 5

# TEST EC PRACTICA 4

1. En la práctica de la bomba, el segundo ejercicio consistía en crear un ejecutable sin “explosiones”, para lo cual se puede utilizar... (marcar la opción \*falsa\*)

- > a) objdump
- b) ddd
- c) hexedit
- d) gdb

2. Una de las “bombas” utiliza el siguiente bucle para cifrar la cadena con la contraseña introducida por el usuario:

80485bb: rolb \$0x4,(%eax)

80485be: add \$0x1,%eax

80485c1: cmp %edx,%eax

80485c3: jne 80485bb <encrypt+0x20>

La intrucción rolb rota el byte destino hacia la izquierda tantos bits como indica el operando fuente. Si inicialmente eax apunta a la cadena del usuario, que se compara con otra cadena

"\x16\x26\x27\x16\x36\x16\x46\x16\x26\x27\x16", almacenada en el código, la contraseña es:

- > a) "\x61\x62\x72\x61\x63\x61\x64\x61\x62\x72\x61" ("abracadabra")
- b) "\x63\x61\x64\x61\x62\x72\x61\x62\x72\x61" ("cadabraabla")
- c) "\x61\x62\x72\x61\x61\x62\x72\x61\x63\x61\x64" ("abraabracad")
- d) "\x61\x72\x62\x61\x64\x61\x63\x61\x72\x62\x61" ("arbadacarba")

3. En una bomba como las estudiadas en prácticas, del tipo...

0x0804873f <main+207>: call 0x8048504 <scanf>

0x08048744 <main+212>: mov 0x24(%esp),%edx

0x08048748 <main+216>: mov 0x804a044,%eax

0x0804874d <main+221>: cmp %eax,%edx

0x0804874f <main+223>: je 0x8048756 <main+230>

0x08048751 <main+225>: call 0x8048604 <boom>

0x08048756 <main+230>: ...

la contraseña es...

- a) el entero 0x804a044
- > b) el entero almacenado a partir de la posición de memoria 0x804a044
- c) el string almacenado a partir de la posición de memoria 0x24(%esp)
- d) ninguna de las anteriores

**4. En la práctica de la bomba necesitamos estudiar el código máquina de la bomba del compañero. A veces dicho código no se visualiza directamente en el depurador ddd, y algunas de las técnicas que se pueden probar para conseguir visualizarlo son... (marcar la opción \*falsa\*)**

- a) comprobar que está activado el panel View → Machine Code Window
- > b) recompilar con información de depuración, por si se nos había olvidado, ya que sin -g el ejecutable no contiene información de depuración
- c) asegurarse de que se ha escrito correctamente el nombre del ejecutable
- d) escribir info line main en el panel de línea de comandos gdb

**5. En la práctica de la bomba, el tercer ejercicio consistía en usar un editor hexadecimal para crear un ejecutable sin "explosiones". Para saber qué contenidos del fichero hay que modificar, se puede utilizar... (marcar la opción \*falsa\*)**

- a) objdump
- b) ddd
- > c) hexedit
- d) gdb

**6. En una bomba como las estudiadas en prácticas, del tipo...**

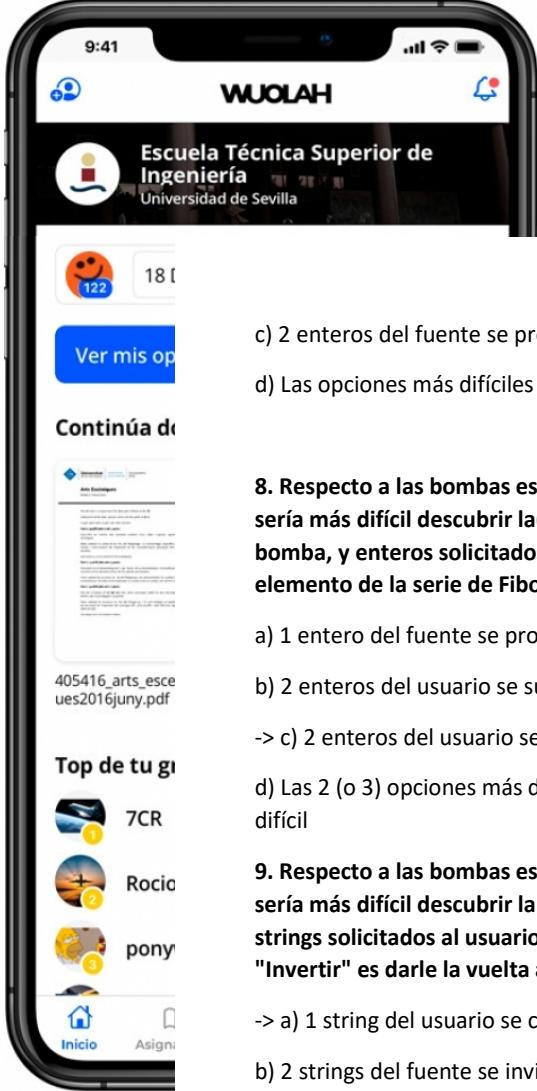
```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>:
```

**...el código numérico (pin) es...**

- > a) el entero almacenado a partir de la posición de memoria 0x804a044
- b) el entero 0x804a044
- c) el entero cuya dirección está almacenada en la posición de memoria 0x804a044
- d) el entero almacenado a partir de la posición de memoria 0x24(%esp)

**7. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la contraseña? Se distingue entre enteros definidos en el código fuente de la bomba, y enteros solicitados al usuario mediante scanf(). Por "procesar" se entiende calcular el n-ésimo elemento de la serie de Fibonacci.**

- > a) 1 entero del usuario se procesa, y se compara con el entero del fuente
- b) 2 enteros del fuente se suman, se procesa la suma, y se compara el resultado con el entero del usuario



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

c) 2 enteros del fuente se procesan, se suman los resultados, y se compara la suma con el entero del usuario

d) Las opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

Continúa dí

**8. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la(s) contraseña(s)? Se distingue entre enteros definidos en el código fuente de la bomba, y enteros solicitados al usuario mediante scanf(). Por "procesar" podemos entender calcular el n-ésimo elemento de la serie de Fibonacci, por ejemplo.**

a) 1 entero del fuente se procesa, y se compara con el entero del usuario

b) 2 enteros del usuario se suman, se procesa la suma, y se compara con el entero del fuente

-> c) 2 enteros del usuario se procesan, se suman los resultados, y se compara con el entero del fuente

d) Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

**9. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la contraseña? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" podemos entender la cifra del César, por ejemplo. "Invertir" es darle la vuelta al string de manera que la primera letra se convierta en la última y viceversa.**

-> a) 1 string del usuario se cifra, y se compara con el string del fuente

b) 2 strings del fuente se invierten, se concatenan, se cifra el resultado, y se compara con el string del usuario

c) 2 strings del fuente se concatenan, se invierte el resultado, se cifra, y se compara con el string del usuario

d) Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

**10. En la práctica de la bomba, el primer ejercicio consistía en “saltarse” las “explosiones”, para lo cual se puede utilizar...**

a) objdump o gdb

b) ddd o hexedit

c) hexedit u objdump

-> d) gdb o ddd

un depurador cualquiera sirve para forzar que se cumpla la condición para “saltarse” la bomba

**12. Una de las “bombas” utiliza el siguiente código para cifrar la clave numérica introducida por el usuario y ahora almacenada en eax:**

**804870d: xor \$0xffff,%eax**

**8048712: mov \$0x2,%ecx**

**8048717: cltd**

**8048718: idiv %ecx**

**804871a: cmp %eax,0x804a034**

Si el entero almacenado a partir de 0x804a034 es 0x7ff, la clave numérica puede ser:

a) 0x1009 4F97 (269 045 655)

b) 0xffff (4095)

**WUOLAH**

c) 0x7ff (2047)

-> d) 1

**11. Para corregir la práctica “bomba digital”, un profesor dispone de 26 ejecutables (y la lista de claves correspondientes) para asignar en el Laboratorio una bomba distinta a cada estudiante. Cuando un estudiante diga que la ha resuelto el profesor exigirá ver al estudiante ejecutando la bomba y tecleando la contraseña y el pin correctos para comprobar que no explota, para así anotarla como resuelta y que le puntúe al estudiante.**

**Un estudiante dice que ha resuelto su bomba, y cuando el profesor pide que se tecleen las claves, la contraseña coincide con la que tiene anotada el profesor, pero el pin no, y de todas formas la bomba no explota. Debería hacerse lo siguiente:**

- a) la bomba tiene que puntuarle al estudiante porque no ha explotado, y el profesor estaría quedando en ridículo (por injusto) si intenta entorpecer o dificultar de cualquier forma que el estudiante se lleve la merecida nota
- > b) el profesor puede pedirle que vuelva a descargar la bomba original e intente repetir con ese ejecutable las claves que acaba de teclear: en realidad eso es sólo para asegurarse completamente de que no se trata de una errata en la lista, aunque también es por la curiosidad de ver si el estudiante se da cuenta de que el profesor sabe lo que ha hecho
- c) hay que denunciar al profesor a los representantes de estudiantes

d) el enunciado es capcioso: si la lista no tiene erratas y el pin se escribe mal la bomba tiene que explotar, y si la bomba no explota es porque las claves se escribieron bien y la lista está mal. En el primer caso no debe puntuar, en el segundo caso sí debe puntuar, y en ningún caso el profesor tiene derecho a hacer comprobaciones distintas a las dos mencionadas

**13. ¿Para qué se utiliza la función gettimeofday en la práctica de la "bomba digital"?**

- a) Para cronometrar y poder comparar las duraciones de las distintas soluciones del programa
- b) Para cifrar la contraseña en función de la hora actual
- > c) Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la contraseña o el PIN
- d) Para imprimir la hora en la pantalla

**14. En una bomba como las estudiadas en prácticas, del tipo...**

0x080486e8 <main+120>: call 0x8048524 <strncmp>

0x080486ed <main+125>: test %eax,%eax

0x080486ef <main+127>: je 0x80486f6 <main+134>

0x080486f1 <main+129>: call 0x8048604 <boom>

0x080486f6 <main+134>: ...

**la contraseña es...**

- a) el valor que tenga %eax
- b) el string almacenado a partir de donde apunta %eax
- c) el entero almacenado a partir de donde apunta %eax
- > d) ninguna de las anteriores

**15. Para corregir la práctica “bomba digital”, un profesor dispone de 26 ejecutables (y la lista de claves correspondientes) para asignar en el Laboratorio una bomba distinta a cada estudiante. Cuando un estudiante diga que la ha resuelto el profesor exigirá ver al estudiante ejecutando la bomba y tecleando la contraseña y el pin correctos para comprobar que no explota, para así anotarla como resuelta y que le puntúe al estudiante.**

**Un estudiante (usando ordenador del Laboratorio con Ubuntu 10.04) dice que ha resuelto su bomba, y cuando el profesor pide que se tecleen las claves, el ddd se “bloquea” y le empieza a “parpadear” al estudiante. Para poder puntuar, el estudiante debería...**

- a) teclear las claves (contraseña y pin), aunque ddd esté “bloqueado” y “parpadeando”
- b) probar en orden los remedios básicos: pulsar <Ctrl>-C varias veces, pulsar <Enter> repetidamente, comprobar que está seleccionada “Machine Code Window”, teclear “info line main”, y si todo falla, ejecutar “rm -rf ~/ .ddd”
- > c) matar la ventana ddd, abrir un terminal con un shell, ejecutar la bomba desde línea de comandos y teclear las claves
- d) reinstalar un paquete ddd más actualizado usando “sudo apt-get install”

**16. En una bomba como las estudiadas en prácticas, del tipo...**

**0x08048705 <main+149>: call 0x80484c4 <gettimeofday>**

...

**0x08048718 <main+168>: cmp \$0x5,%eax**

**0x0804871b <main+171>: jle 0x8048722 <main+178>**

**0x0804871d <main+173>: call 0x8048604 <boom>**

**0x08048722 <main+178>: ...**

**ejecutada paso a paso con el depurador ddd, interesaría...**

- a) ejecutar hasta jle, ajustar %eax a 6, y continuar ejecutando paso a paso
- b) ejecutar hasta jle, ajustar %eax a 4, y continuar ejecutando paso a paso
- > c) cambiar jle por jmp usando ddd o un editor hex, salvar el programa y reiniciar la depuración con el nuevo ejecutable
- d) Ninguna de las opciones anteriores es de interés

**17. En una bomba como las estudiadas en prácticas, del tipo...**

**0x080486e8 <main+120>: call 0x8048524 <strncmp>**

**0x080486ed <main+125>: test %eax,%eax**

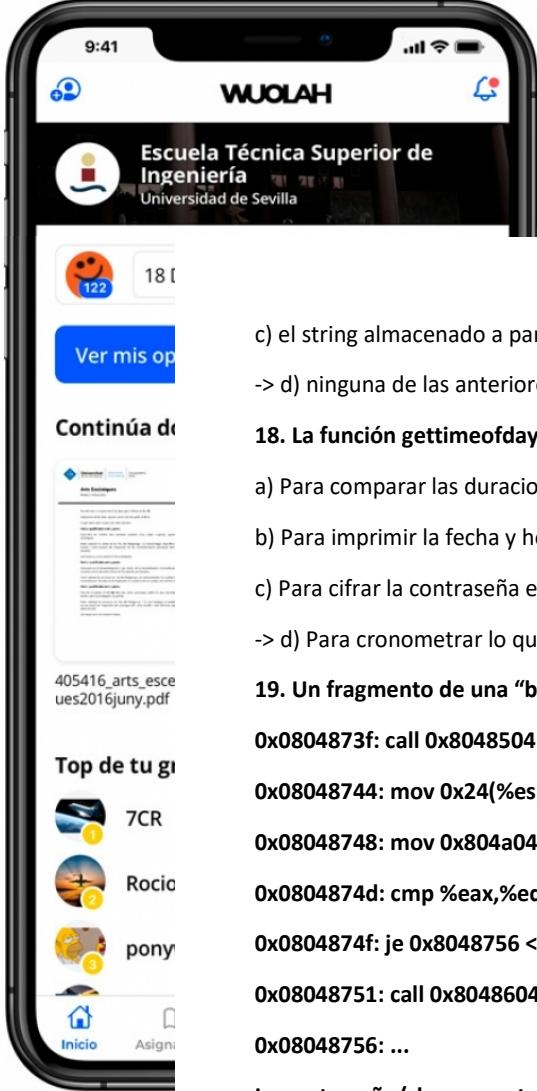
**0x080486ef <main+127>: je 0x80486f6 <main+134>**

**0x080486f1 <main+129>: call 0x8048604 <boom>**

**0x080486f6 <main+134>: ...**

**la contraseña es...**

- a) el string almacenado a partir de 0x8048524
- b) el string almacenado a partir de 0x80486f6



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

c) el string almacenado a partir de 0x8048604

-> d) ninguna de las anteriores

## 18. La función gettimeofday() en la práctica de la "bomba digital" se utiliza para:

a) Para comparar las duraciones de las distintas soluciones del programa

b) Para imprimir la fecha y hora

c) Para cifrar la contraseña en función de la hora actual

-> d) Para cronometrar lo que tarda el usuario en introducir la contraseña

## 19. Un fragmento de una "bomba" desensamblada es:

0x0804873f: call 0x8048504 <scanf>

0x08048744: mov 0x24(%esp),%edx

0x08048748: mov 0x804a044,%eax

0x0804874d: cmp %eax,%edx

0x0804874f: je 0x8048756 <main+230>

0x08048751: call 0x8048604 <boom>

0x08048756: ...

La contraseña/clave en este caso es...

a) el string almacenado a partir de la posición de memoria 0x804a044

b) el string almacenado a partir de la posición de memoria 0x24(%esp)

-> c) el entero almacenado a partir de la posición de memoria 0x804a044

d) el entero 0x804a044

## 20. En la realización de la práctica de la bomba digital, una parte del código máquina es el siguiente:

0x080486e8 <main+120>: call 0x8048524 <strncpy>

0x080486ed <main+125>: test %eax,%eax

0x080486ef <main+127>: je 0x80486f6 <main+134>

0x080486f1 <main+129>: call 0x8048604 <boom>

¿Cuál de los siguientes comandos cambiaría el salto condicional por un salto incondicional?

a) set %0x080486ef=0xeb

-> b) set \*(char\*)0x080486ef=0xeb

c) set \$0x080486ef=0xeb

d) set \*(char\*)0x080486f6=jmp

**21. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más fácil descubrir la(s) contraseña(s)? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" se entiende aplicar la cifra del César (sumar o restar una constante fija a los códigos ASCII).**

- > a) 1 string del fuente se cifra, y se compara con el string del usuario
- b) 2 strings del usuario se concatenan, se cifra el resultado y se compara con el string del fuente
- c) 2 strings del usuario se cifran, se concatenan los resultados, y se compara con el string del fuente
- d) Las opciones más fáciles son de la misma dificultad, así que no se puede marcar ninguna como la más fácil

**22. En una bomba como las estudiadas en prácticas, del tipo...**

0x080486e8 <main+120>: call 0x8048524 <strncmp>

0x080486ed <main+125>: test %eax,%eax

0x080486ef <main+127>: je 0x80486f6 <main+134>

0x080486f1 <main+129>: call 0x8048604 <boom>

0x080486f6 <main+134>:

**...la contraseña es...**

- a) el valor que tenga %eax
- b) el string almacenado a partir de 0x80486f6
- c) el entero almacenado a partir de donde apunta %eax
- > d) ninguna de las anteriores

**23. ¿Para qué se utiliza la función scanf() en la práctica de la "bomba digital"?**

- a. Para escanear el fichero ejecutable “bomba” y asegurarse de que no contenga virus
- b. Para leer la contraseña tecleada por el usuario
- > c. Para leer el PIN o código numérico //P4 pág.3 Fig.1
- d. Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la contraseña o el PIN

**24. En una bomba como las estudiadas en prácticas, del tipo...**

0x40079b <main+64> lea 0x30(%rsp),%rdi

0x4007a0 <main+69> mov 0x2008d9(%rip),%rdx # 0x601080

0x4007a7 <main+76> mov \$0x64,%esi

0x4007ac <main+81> callq 0x400600 <fgets@plt>

0x4007b1 <main+86> test %rax,%rax

0x4007b4 <main+89> je 0x400785 <main+42>

0x4007b6 <main+91> lea 0x30(%rsp),%rdi

0x4007bb <main+96> mov \$0xd,%edx

0x4007c0 <main+101> lea 0x2008a1(%rip),%rsi # 0x601068

```

0x4007c7 <main+108> callq 0x4005d0 <strncmp@plt>
0x4007cc <main+113> test %eax,%eax
0x4007ce <main+115> je 0x4007d5 <main+122>
0x4007d0 <main+117> callq 0x400727 <boom>
0x4007d5 <main+122> lea 0x20(%rsp),%rdi
...la contraseña (alfanumérica) es...
a. el string almacenado a partir de 0x601068
b. el string almacenado a partir de 0x2008a1+0x4007c7
c. el string almacenado a partir de 0x30(%rsp)
-> d. no se puede marcar una y sólo una de las respuestas anteriores
25. En una bomba como las estudiadas en prácticas, del tipo...
0x40080f <main+180> lea 0xc(%rsp),%rsi
0x400814 <main+185> lea 0x1dd(%rip),%rdi # 0x4009f8
0x40081b <main+192> mov $0x0,%eax
0x400820 <main+197> callq 0x400620 <__isoc99_scanf@plt>
0x400825 <main+202> mov %eax,%ebx
0x400827 <main+204> test %eax,%eax
0x400829 <main+206> jne 0x40083c <main+225>
0x40082b <main+208> lea 0x1c9(%rip),%rdi # 0x4009fb
0x400832 <main+215> mov $0x0,%eax
0x400837 <main+220> callq 0x400620 <__isoc99_scanf@plt>
0x40083c <main+225> cmp $0x1,%ebx
0x40083f <main+228> jne 0x4007f9 <main+158>
0x400841 <main+230> mov 0x200819(%rip),%eax # 0x601060
0x400847 <main+236> cmp %eax,0xc(%rsp)
0x40084b <main+240> je 0x400852 <main+247>
0x40084d <main+242> callq 0x400727 <boom>
0x400852 <main+247> lea 0x10(%rsp),%rdi
...el código numérico (pin) es...
a. el entero 0x601060
b. el entero cuya dirección está almacenada en la posición de memoria 0x4009f8
c. el entero almacenado a partir de la posición de memoria 0x4009fb
-> d. el entero almacenado a partir de la posición de memoria 0x200819+0x400847

```

# Prácticas tema 6 EC

**1. En la práctica de la cache, el código de “size.cc” accede al vector saltando de 64 en 64. ¿Por qué?**

- a) Porque cada elemento del vector ocupa 64 bytes
- b) Para recorrer el vector más rápidamente
- c) Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB
- > d) Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal

**2. ¿Cuál de las siguientes afirmaciones sobre las caches es \*FALSA\*?**

- a) La cache de nivel 3 no contiene toda la memoria que maneja el programa
- b) Las direcciones a las que accede un programa no son completamente aleatorias, sino que se rigen por ciertos patrones de localidad
- c) Un procesador actual tiene varias caches de nivel 1
- > d) Casi ningún procesador actual tiene memoria cache L2

**3. ¿Cuál de las siguientes afirmaciones sobre el programa size.cc de la práctica 5 es cierta?**

- a) La diferencia de velocidades entre L2 y L3 es mayor que la diferencia de velocidades entre L1 y L2.

puede que haya sistemas con esa característica, pero desde luego los que usamos nosotros en prácticas no.

- b) Si continuáramos multiplicando por 2 el tamaño del vector en el eje X obteniendo más puntos de la gráfica, esta continuaría horizontal para cualquier valor más allá de 64 MB.

-> c) La gráfica tiene escalones hacia arriba porque en cada punto del eje X accedemos al mismo número de elementos del vector y el número de aciertos por localidad temporal disminuye bruscamente en ciertos puntos al aumentar el tamaño del vector.

si cabe el vector en cache, los millones de accesos (siempre el mismo número de accesos) son todo aciertos por localidad temporal

- d) La gráfica tiene tramos horizontales porque el hecho de realizar la mitad de accesos al vector en cada punto de un tramo horizontal respecto al anterior punto de ese mismo tramo horizontal es compensado por el número de fallos creciente en ese mismo tramo horizontal.

**4. El servidor de SWAD tiene dos procesadores Xeon E5540 con 4 núcleos cada uno. Cada procesador tiene 4 caches L1 de instrucciones de 32 KB, 4 caches L1 de datos de 32 KB, 4 caches unificadas L2 de 256 KB y una cache unificada L3 de 8MB. Suponga que un proceso swad, que se ejecuta en un núcleo, tiene que ordenar un vector de estudiantes accediendo repetidamente a sus elementos. Cada elemento es una estructura de datos de un estudiante y tiene un tamaño de 4KB. Si representamos en una gráfica las prestaciones en función del número de estudiantes a ordenar, ¿para qué límites teóricos en el número de estudiantes se observarán saltos en las prestaciones debidos a accesos a la jerarquía de memoria?**

- a) 32 / 256 / 8192 estudiantes
- b) 16 / 32 / 64 estudiantes
- > c) 8 / 64 / 2048 estudiantes
- d) 4 / 32 / 512 estudiantes

**5. En el programa "size" de la práctica de la cache, si el primer escalón pasa de tiempo = 1 para todos los tamaños de vector menores o iguales que 32 KB a tiempo = 3 para los tamaños 64 KB y 128 KB, podemos asegurar que:**

**(Nota: en las opciones de respuesta, "rapidez" se refiere a  $T_i$ , el tiempo de acceso efectivo al nivel  $i$ , no a  $t_i$ , el tiempo de acceso propio al nivel  $i$ . Si dice "La cache L1 es 3x más rápida que L2", se refiere a que  $T_1=T_2/3$ , no a que  $t_1=t_2/3$ . Ver T6 tr.26)**

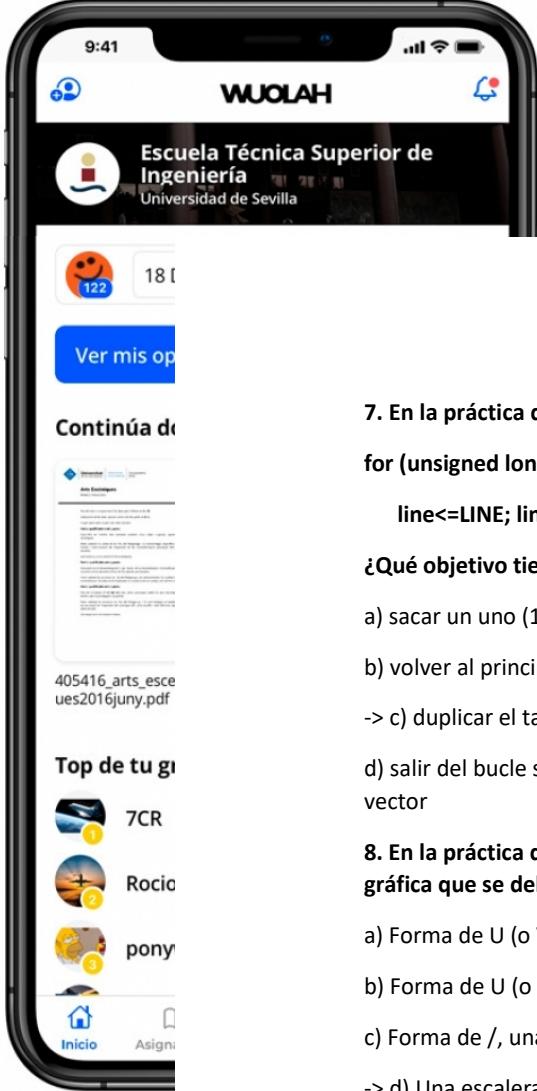
- 
- a) la cache L2 es como mucho el doble de rápida que la memoria principal
  - b) la cache L2 es al menos el doble de rápida que la memoria principal
  - c) la cache L1 es como mucho tres veces más rápida que la cache L2
  - > d) la cache L1 es al menos tres veces más rápida que la cache L2

sólo sería exactamente 3 si el tiempo de cómputo adicional a los accesos de memoria fuera despreciable. Como en general no lo es, los accesos deben ser aún más rápidos para que el incremento sea 3x.

Matemáticamente, el enunciado dice que  $c+m_1=1$ ,  $c+m_2=3$ , entonces  $(c+m_2)/(c+m_1)=3$ , y despejando sale  $m_1=(m_2)/3-2c/3$ , es decir, sólo si  $c=0$  entonces  $m_1=(m_2)/3$ , y si  $c>0$  entonces  $m_1<(m_2)/3$

**6. Suponer una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponer que para la siguiente secuencia de direcciones enviadas a la cache: 0, 2, 4, 8, 16, 32, la tasa de acierto es 0.33. ¿Cuál es el tamaño de bloque de la cache?**

- a) 4 bytes
- > b) 8 bytes
- c) 16 bytes
- d) Ninguno de los anteriores



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

**7. En la práctica de la cache, el código de line.cc incluye la sentencia**

```
for (unsigned long long line=1;
 line<=LINE; line<<=1) { ... }
```

**¿Qué objetivo tiene la expresión line<<1?**

- a) sacar un uno (1) por el stream line
- b) volver al principio del vector cuando el índice excede la longitud del vector
- > c) duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior
- d) salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector

**8. En la práctica de cache hemos hecho una gráfica con el código size.cc ¿Qué forma tiene la gráfica que se debe obtener?**

- a) Forma de U (o V) con un tramo descendente y otro ascendente
- b) Forma de U (o V) invertida, con un tramo ascendente y otro descendente
- c) Forma de /, una gráfica siempre creciente y sin escalones
- > d) Una escalera con varios tramos horizontales

**9. Abajo se ofrece el listado de una función para multiplicar matrices C = A x B.**

```
void mult_matr(float A[N][N], float B[N][N], float C[N][N]){
 /* Se asume valor inicial C = {0,0,...} */

 int i,j,k;

 for (i=0; i<N; i++)
 for (j=0; j<N; j++)
 for (k=0; k<N; k++)
 C[i][j] += A[i][k] * B[k][j];
}
```

**Suponer que:**

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.
- El tamaño de los tipos de datos es como en IA32.
- El compilador optimiza el acceso a C[i][j] en un registro.

**Imaginar que se modifica la última sentencia (el cuerpo anidado) por esta otra**

```
C[i][j] += A[i][k] * B[j][k];
```

**de manera que se calcule  $C = A \times B'$  ( $A$  por traspuesta de  $B$ ). Aproximadamente, ¿qué tasa de fallos se podría esperar de esta nueva función para valores grandes de  $N$ ?**

- a) 1/8
- b) 1/4
- c) 1/2
- > d) 1/16

uno de cada 16  $A[i][k]$  y otro de cada 16  $B[i][k]$

**10. En el programa “size” de la práctica de la cache, si el primer escalón pasa de tiempo=2 para un tamaño de vector menor que 32KB a tiempo=8 para un tamaño mayor que 32KB, podemos asegurar que:**

---

**(Nota: en las opciones de respuesta, "rapidez" se refiere a  $T_i$ , el tiempo de acceso efectivo al nivel  $i$ , no a  $t_i$ , el tiempo de acceso propio al nivel  $i$ . Si dice "La cache L1 es 4x más rápida que L2", se refiere a que  $T_1=T_2/4$ , no a que  $t_1=t_2/4$ . Ver T6 tr.26)**

---

- a) La cache L1 es como mucho cuatro veces más rápida que la cache L2
- b) La cache L1 es seis veces más rápida que la cache L2
- > c) La cache L1 es al menos cuatro veces más de rápida que la cache L2

sólo sería exactamente 4 si el tiempo de cómputo adicional a los accesos de memoria fuera despreciable. Como en general no lo es, los accesos deben ser aún más rápidos para que el incremento sea 4x.

Matemáticamente, el enunciado dice que  $c+m_1=2$ ,  $c+m_2=8$ , entonces  $(c+m_2)/(c+m_1)=4$ , y despejando sale  $m_1=(m_2)/4-3c/4$ , es decir, sólo si  $c=0$  entonces  $m_1=(m_2)/4$ , y si  $c>0$  entonces  $m_1<(m_2)/4$

- d) La cache L1 es cuatro veces más rápida que la cache L2

**11. En la práctica de la cache, el código de size.cc accede al vector saltando de 64 en 64. ¿Por qué?**

- a) Porque cada elemento del vector ocupa 64 bytes
- b) Para recorrer el vector más rápidamente
- c) Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB
- > d) Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal

**12. Sea un computador de 32 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:**

```
int v[262144];
for (i = 0; i < 262144; i += 8)
 v[i] = 9;
```

**¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?**

- a) 0 (ningún fallo)
- > b) 1/2 (mitad aciertos, mitad fallos)

es un array de ints (4B) y se salta  $i+=8$ , los accesos están separados 32B y las líneas son de 64B, los accesos con índice  $i$  par son fallos, con índice impar son aciertos.

- c) 1/8 (un fallo por cada 8 accesos)
- d) 1 (todo son fallos)

**13. En la práctica de la cache, el código de “line.cc” incluye la sentencia**

```
for (unsigned long long line=1; line<=LINE; line<<=1) { ... }
```

**¿Qué objetivo tiene la expresión  $line<<=1$ ?**

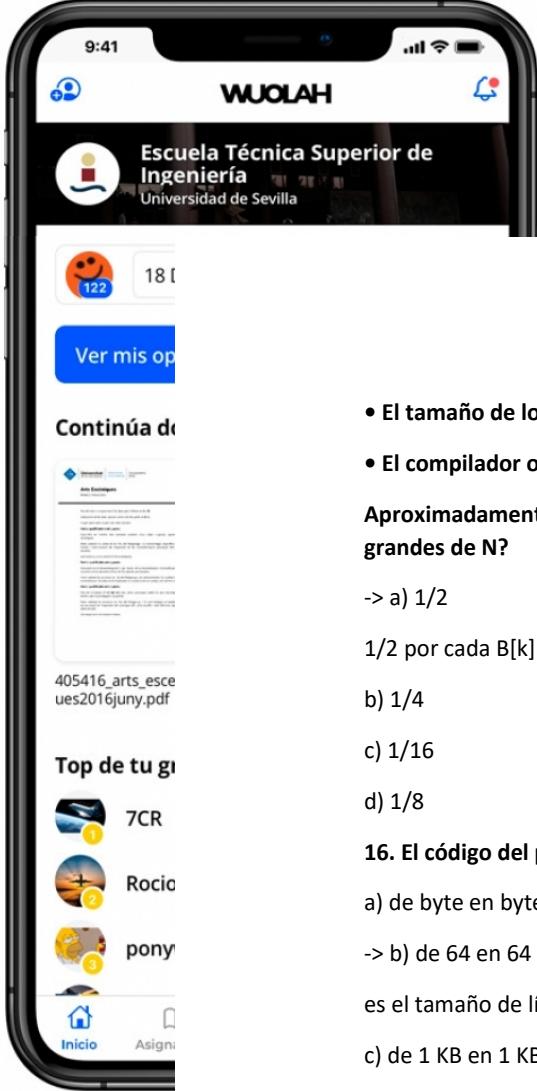
- a) sacar un uno (1) por el stream line
- b) volver al principio del vector cuando el índice exceda la longitud del vector
- c) salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
- > d) duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior

**15. Abajo se ofrece el listado de una función para multiplicar matrices  $C = A \times B$ .**

```
void mult_matr(float A[N][N], float B[N][N], float C[N][N]){\n/* Se asume valor inicial C = {0,0...} */\n\nint i,j,k;\n\nfor (i=0; i<N; i++)\n for (j=0; j<N; j++)\n for (k=0; k<N; k++)\n C[i][j] += A[i][k] * B[k][j];\n}
```

**Suponer que:**

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

- El tamaño de los tipos de datos es como en IA32.

- El compilador optimiza el acceso a  $C[i][j]$  en un registro.

Aproximadamente, ¿qué tasa de fallos se podría esperar de esta función para valores grandes de N?

-> a) 1/2

1/2 por cada  $B[k][j]$  + 1/32 por cada 16-ésimo  $A[i][k]$

b) 1/4

c) 1/16

d) 1/8

**16. El código del programa "size" de la práctica de la cache accede al vector saltando...**

a) de byte en byte

-> b) de 64 en 64 bytes

es el tamaño de línea deducido en la práctica "line"

c) de 1 KB en 1 KB

d) de 64 KB en 64 KB

**18. En el programa line.cc de la práctica 5, si para cada tamaño de línea (line) recorremos una única vez el vector, la gráfica resultante es decreciente porque:**

a) Cada vez que line aumenta al doble, el número de aciertos por localidad temporal aumenta, porque ya habíamos accedido a cada posición i del vector cuando lo recorrimos en el punto anterior del eje X.

b) Cada vez que line aumenta al doble, el número de aciertos por localidad espacial aumenta, porque ya habíamos accedido a cada posición i-1 del vector cuando lo recorrimos en el punto anterior del eje X.

c) Cada vez que line aumenta al doble, se accede con éxito a más posiciones del vector en niveles de la jerarquía de memoria más rápidos.

-> d) Cada vez que line aumenta al doble, realizamos la mitad de accesos al vector que para el valor anterior.

**19. Sea un computador de 32 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:**

```
int v[262144];
for (i = 0; i < 262144; i += 2)
 v[i] = 9;
```

**¿Cuál será la tasa de fallos aproximada que se obtiene en la primera ejecución del bucle anterior?**

- a) 0 (ningún fallo)
- b) 1/2 (mitad aciertos, mitad fallos)
- > c) 1/8 (un fallo por cada 8 accesos)
- d) 1 (todo son fallos)

# **TEST TEMA 1 EC**

**1. ¿En qué generación, dentro de la historia de los computadores digitales, se alcanzaron tiempos de conmutación del orden de nanosegundos?**

- a) primera
- b) segunda
- >c) tercera
- d) cuarta

**2. En una arquitectura RISC típica:**

- > a) se usa un porcentaje elevado de las instrucciones del repertorio.
- b) no puede usarse segmentación.
- c) la programación resulta mucho más simple que en una arquitectura CISC.
- d) la UC es más compleja que en una arquitectura CISC.

**3. ¿Cuál de las siguientes afirmaciones es incorrecta?**

- a) El repertorio de instrucciones es el conjunto de operaciones que es capaz de interpretar la unidad de control.
- b) El modo de direccionamiento permite determinar un operando o la ubicación del operando.
- > c) Los operandos siempre están almacenados en memoria.
- d) El repertorio de instrucciones debe ser capaz de realizar una tarea en un tiempo finito.

**4. Un modo de direccionamiento en el que se especifica un registro y una dirección de memoria cuyo contenido se suma al contenido del registro base para obtener la dirección efectiva, se conoce como:**

- a) base con desplazamiento
- b) directo o absoluto
- c) indirecto a registro través de memoria
- > d) ninguno de los anteriores

**5. El ancho de palabra de una memoria corresponde a:**

- a) El número que identifica únicamente cada posición de la memoria.
- > b) La cantidad de bits que caben en una sola posición
- c) La longitud del registro de direcciones de la memoria.
- d) El número de posiciones que la componen.

**6. En la captación de la instrucción:**

- a) en MAR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- b) en MBR indicamos la dirección donde está la instrucción y en la ALU recogemos la instrucción.
- > c) en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción.
- d) en MBR indicamos la dirección donde está la instrucción y en MAR recogemos la instrucción.

**7. Un sistema con direcciones de 8bits utiliza una puerta NAND conectada a las líneas A7...A5 para atacar la entrada CS# (activa baja) de un módulo de memoria. En el mapa de memoria las siguientes posiciones corresponderán a dicho módulo**

- a) 0x00 a 0x1f
- b) 0x00 a 0x0f y 0x80 a 0x8f
- > c) 0xe0 a 0xff
- d) 0x70 a 0x7f y 0xf0 a 0xff

**8. ¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?**

- a) primera
- b) segunda
- > c) tercera
- d) cuarta

**9. Una instrucción máquina del tipo "Add M,R" podría formar parte del repertorio de**

- a) una máquina con arquitectura R/R  
sería ADD R1,R2
- b) una máquina pila  
sería ADD
- > c) una máquina con arquitectura M/M  
podría ser también ADD M,M
- d) una máquina de acumulador  
sería ADD M

**10. ¿Cuál de las siguientes no es una característica de los computadores RISC?**

- a) Un computador RISC no debe emplear microprogramación.
- b) Para acelerar el computador RISC se emplean técnicas de pipelining.
- > c) Las funciones que realizan los computadores RISC deben ser lo más complejas y potentes que sea posible.
- d) La decodificación de las instrucciones debe ser simple: un computador RISC debería emplear un único formato de instrucción

**11. ¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción IA32 add array(%ebx,4), %edx?**

- a) Direccionamiento indexado
- > b) Direccionamiento a registro
- c) Direccionamiento relativo a registro base
- d) Direccionamiento inmediato

**12. ¿Qué novedad se desarrolló en la tercera generación de computadores?**

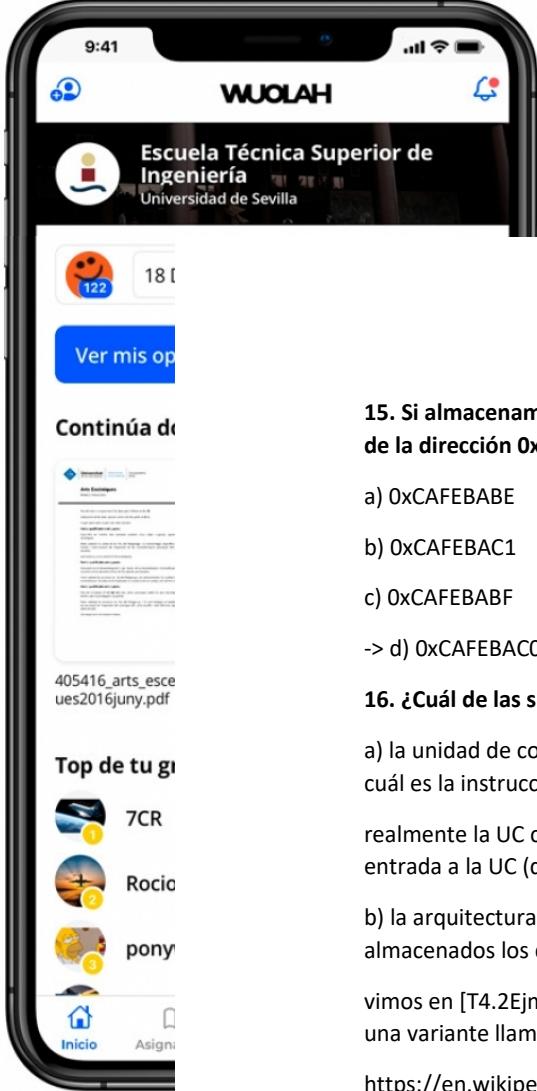
- a) Los microprocesadores CISC
- > b) Los circuitos integrados
- c) Los microprocesadores RISC
- d) Los primeros lenguajes de programación de alto nivel

**13. En un procesador de la familia 80x86 las posiciones de memoria que representan una variable long (entero 4B compl.2) contiene los bytes: F0 FF FF FF. ¿Cuánto vale dicha variable?**

- > a) -16
- b) 16
- c) 4294967280
- d) 4043309055

**14. En el direccionamiento inmediato, tras captarse completamente la instrucción:**

- > a) se accede al operando, que es una constante contenida en la propia instrucción.
- b) el código de operación contiene el operando.
- c) se accede al operando, que está contenido en una posición de memoria principal.
- d) se accede al operando, que se encuentra almacenado en uno de los registros programables.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**15. Si almacenamos según el criterio little-endian la palabra de 64 bits 0xFACEB00C a partir de la dirección 0xCAFEBAE, el byte 0xCE quedará almacenado en la dirección:**

- a) 0xCAFEBAE
- b) 0xCAFEBAE1
- c) 0xCAFEBABF
- > d) 0xCAFEBAE0

**16. ¿Cuál de las siguientes afirmaciones es cierta?**

a) la unidad de control necesita como entrada el registro contador de programa para saber cuál es la instrucción que debe ejecutar a continuación

realmente la UC copia PC en MAR, y lo que lee en MDR (captación) lo lleva a IR que sí es entrada a la UC (decodificación)

b) la arquitectura von Neumann de los computadores tradicionales consiste en tener almacenados los datos separados de las instrucciones en memorias distintas

vimos en [T4.2EjmSeg] tr.12 que tener memoria de instrucciones separada de la de datos es una variante llamada "arquitectura Harvard", llamándose "arquitectura Princeton" la original

[https://en.wikipedia.org/wiki/Von\\_Neumann\\_architecture](https://en.wikipedia.org/wiki/Von_Neumann_architecture)

[https://en.wikipedia.org/wiki/Modified\\_Harvard\\_architecture](https://en.wikipedia.org/wiki/Modified_Harvard_architecture)

c) el registro de direcciones de memoria es un registro de propósito general que puede contener tanto direcciones como datos

MAR no está disponible al programador, lo usa la UC para indicar la dirección de memoria a leer/escribir

-> d) el registro de estado (flags) es un registro de propósito específico cuyo contenido puede ser visto directa o indirectamente por el usuario mediante el uso de ciertas instrucciones específicas

aunque no hemos visto pushf y popf, por eliminación ésta es la respuesta

**17. Un bus se compone de:**

- a) líneas de alimentación
- b) líneas de datos y líneas de dirección
- > c) líneas de control/estado, líneas de dirección y líneas de datos
- d) líneas de estado y líneas de control

**18. ¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:**

push #4

push #7

push #8

add

push #10

sub

mul

a) 4, 7, 48

b) 4

c) 4, 7, 8, 10

-> d) 20

**19. La primera generación de computadores se caracteriza por el uso de:**

a) Microprocesadores.

b) Transistores.

c) Fibra óptica.

-> d) Tubos de vacío.

**20. ¿Cuál de las siguientes afirmaciones sobre el direccionamiento absoluto es falsa?**

a) El objeto está en una posición de la memoria

b) La instrucción contiene la dirección de memoria en la que se encuentra el objeto

-> c) El tamaño del operando direccionado queda limitado por el nº de bits del campo de direccionamiento

d) El rango de posiciones direccionables queda limitado por el tamaño del campo de direccionamiento

No hemos explicado detalladamente que absoluto = directo con dirección completa. Evitar esta pregunta

**21. Son funciones de la unidad de control:**

a) la codificación de las instrucciones máquina

b) la lectura de memoria principal de la instrucción apuntada por el µPC

-> c) el secuenciamiento de las instrucciones máquina

d) todas las respuestas son ciertas

**22. ¿De qué depende el tamaño del contador de programa?**

- a) Del ancho del bus de datos.
- > b) Del número de direcciones de memoria.
- c) Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.
- d) De la longitud del código de operación de las instrucciones.

**23. Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x8600 como:**

- a) M[0x8600]=0x89 y M[0x8601]=0x65
- > b) M[0x8600]=0x65 y M[0x8601]=0x89
- c) M[0x8600]=0x69 y M[0x8601]=0x85
- d) M[0x8600]=0x85 y M[0x8601]=0x69

**24. Una máquina superescalar es aquella que:**

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- > c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
- d) ninguna respuesta de las anteriores es correcta.

**25. Si queremos almacenar la palabra de 64bits 0x0000001f ffffffe0 en una memoria de bytes según la convención little-endian a partir de la posición 0x0804913c, quedará**

- a) 0x00 en 0x0804913c y 0xe0 en 0x08049143
- b) 0x1f en 0x0804913c y 0xe0 en 0x08049140
- > c) 0xe0 en 0x0804913c y 0x1f en 0x08049140
- d) Todas las respuestas anteriores son incorrectas

**26. ¿En qué generación, dentro de la historia de los computadores digitales, aparecen los sistemas operativos multiusuario?**

- a) primera
- b) segunda
- > c) tercera
- d) cuarta

**27. Si queremos almacenar la palabra de 16 bits 0x9660 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x1000 como:**

- a) M[0x1000]=0x96 y M[0x1001]=0x60
- b) M[0x1000]=0x69 y M[0x1001]=0x06
- > c) M[0x1000]=0x60 y M[0x1001]=0x96
- d) M[0x1000]=0x06 y M[0x1001]=0x69

**28. Para obtener una única velocidad comparativa final, el benchmark SPEC CPU combina las velocidades de ejecución de una serie de tests, respecto a un ordenador de referencia, usando la media...**

- a) armónica
- b) aritmética
- c) ponderada
- > d) geométrica

**29. Un procesador con E/S mapeada a memoria tiene un bus de direcciones de 10 líneas y uno de datos de 8. El mapa de memoria tiene 512 posiciones para código (ROM), 256 para datos (RAM) y 256 para E/S, en ese orden. Los rangos de direcciones para esas tres zonas serán:**

- a) 000 a 7FF, 800 a BFF y C00 a FFF
- > b) 000 a 1FF, 200 a 2FF y 300 a 3FF
- c) 000 a 9FF, A00 a CFF y D00 a FFF
- d) 000 a 5FF, 600 a 7FF y 800 a 9FF

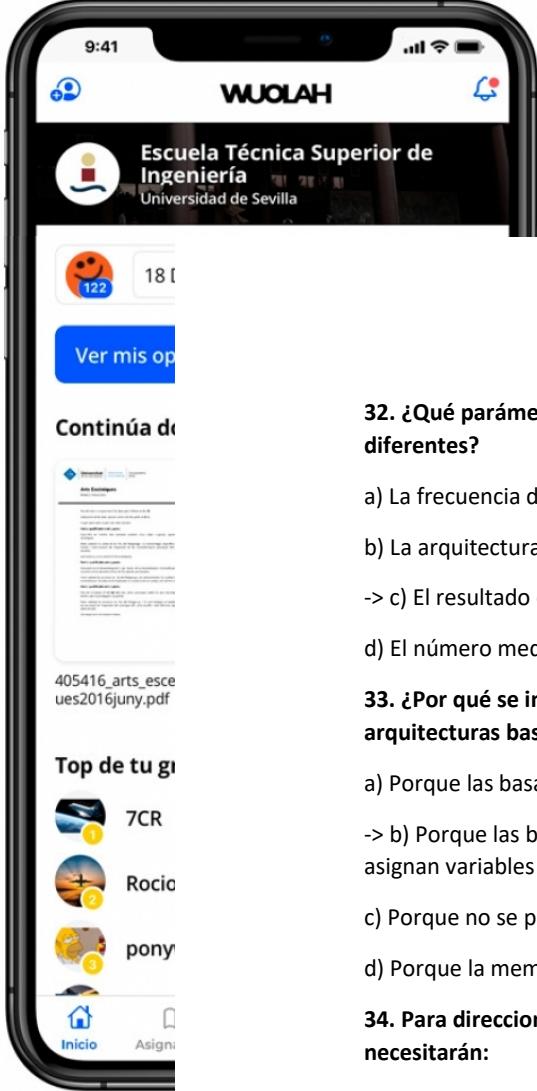
**30. ¿Cuál de las siguientes direcciones NO está alineada a double (8-byte)?**

(Al no poder escribir el 2 como subíndice, aclaramos que ")2" indica binario)

- > a) 1110110101110100)2
- b) 1110110101101000)2
- c) 1110110101110000)2
- d) Todas están alineadas a doblé

**31. En una arquitectura RISC típica:**

- > a) suele usarse segmentación
- b) la programación resulta mucho más simple que en una arquitectura CISC
- c) se usan pocas instrucciones de las disponibles en el conjunto de instrucciones
- d) la UC es más compleja que en una arquitectura CISC



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**32. ¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?**

- a) La frecuencia de reloj del procesador.
- b) La arquitectura del procesador.
- > c) El resultado de la ejecución de un conjunto de programas de prueba.
- d) El número medio de ciclos de reloj por instrucción.

**33. ¿Por qué se impusieron las arquitecturas de registros de propósito general a las arquitecturas basadas en pila?**

- a) Porque las basadas en registros permiten reducir el tamaño del programa
- > b) Porque las basadas en registros son capaces de lograr un mejor rendimiento cuando se asignan variables a registros
- c) Porque no se puede programar una arquitectura de pila en un lenguaje de alto nivel
- d) Porque la memoria es más cara que los registros

**34. Para direccionar una memoria de bytes en la que quepan 2G palabras de 32 bits se necesitarán:**

- a) 21 bits como máximo
- b) 32 bits exactamente
- > c) 33 bits como mínimo
- d) 31 bits como mínimo

**35. ¿Cuál de los siguientes no es un tipo de bus?**

- > a) Secuencial
- opuestos a buses paralelos son los buses serie
- opuestos a programas secuenciales son los programas paralelos
- b) Sistema
- c) E/S
- d) Paralelo

**36. En una CPU de 32 bits con memoria de bytes, el problema es que...**

- a) No tiene sentido, un registro no cabría en memoria
- b) Hay que usar 4 instrucciones de lectura (o escritura) para leer (o escribir) un registro completo
- > c) Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU
- d) No hay problema, cuando se salva un registro a memoria se escribe en la posición deseada

**37. ¿Cuál es la característica tecnológica principal de la tercera generación de computadores?**

- a) Las válvulas
- b) Los transistores
- c) La gran integración de los circuitos (VLSI)
- > d) Los circuitos integrados

**38. ¿Cuál es el valor mínimo (más negativo) que puede tomar un entero de 32bits en complemento a dos?**

( $\wedge$  simboliza potenciación)

- a)  $-2^{31} + 1$
- b)  $-2^{32} + 1$
- > c)  $-2^{31}$

Present. tr.5,12, Intro. tr.8

- d)  $-2^{32}$

**39. En las instrucciones aritméticas con dos operandos de un procesador con arquitectura de pila, los dos operandos...**

- a) son dos registros del procesador.
- b) pueden estar en cualquier posición de la pila.
- c) se introducen en la pila tras realizar la operación.
- > d) son la cima de la pila y el elemento siguiente de la cima de la pila.

**40. El direccionamiento directo a memoria utiliza...**

- a) dos desplazamientos contenidos en la propia instrucción.
- b) un registro y un desplazamiento contenidos en la propia instrucción.
- > c) un desplazamiento.
- d) un registro.

**41. En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1 B), media palabra (2 B) y palabra (4 B), se almacenan a partir de la posición 0xCAFEBA0 cuatro palabras con valores -1, -2, -3, -4. ¿Qué se obtendría al consultar la media palabra de la posición 0xCAFEBABE?**

- > a) -1

los contenidos son

CAFEBA0: FF FF FF FF

CAFEBAB4: FE FF FF FF

CAFEBAB8: FD FF FF FF

CAFEBABC: FC FF FF FF

las últimas dos posiciones, a partir de CAFEBABC, contienen FF FF, que es -1

b) -4

c) no se puede saber, faltan datos

d) ninguna de las anteriores

**42. ¿Cuál de las siguientes afirmaciones sobre el benchmark SPEC CPU es falsa?**

-> a) El resultado final es la media aritmética de las (12 ó 17) velocidades, bien sea de enteros ó de punto flotante (SPECint2006 ó SPECfp2006)

es la media geométrica

b) Se cronometran unos 12 tests de enteros (CINT2006) y unos 17 tests de punto flotante (CFP2006)

c) La última versión es SPEC CPU2006 V1.2 de 2011

d) Se usa como referencia un computador UltraSPARC II 300MHz, y para cada test se calcula el cociente entre el tiempo de ejecución en el computador a testear y en el de referencia

**43. Se pretende almacenar una palabra de 4 B en una memoria de bytes a partir de una dirección determinada. ¿Cuál de las siguientes es válida, si la palabra debe quedar alineada?**

-> a) 0xFACEB00C

b) 0xCAFEBAE

c) 0xABADFOOD

d) 0xDEADBEEF

**44. ¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria virtual?**

a) primera

b) segunda

-> c) tercera

d) cuarta

**45. ¿En qué pareja de registros están el dato/instrucción que se leerá o escribirá en memoria, y la dirección de memoria?**

-> a) MBR y MAR

b) MAR y ACUMULADOR

c) MBR y PC

d) IR y ACUMULADOR

**46. La idea de desarrollar máquinas CISC surgió para:**

- a) simplificar el diseño hardware de la UC.
- > b) tener instrucciones cercanas al lenguaje de alto nivel.
- c) conseguir un conjunto de instrucciones cortas y sencillas de decodificar.
- d) ninguna de las respuestas anteriores es cierta.

**47. Un modo de vídeo de 512 x 256 píxeles y 16 colores por píxel ocupa una memoria de:**

- a) 2 MB
- > b) 64 KB
- c) 128 KB
- d) 8 KB

**48. ¿Cuál de las siguientes direcciones está alineada a double (8-byte)?**

(Al no poder escribir el 2 como subíndice, aclaramos que ")2" indica binario)

- a) 1110110101110111)2
- b) 1110110101110100)2
- > c) 1110110101110000)2
- d) Ninguna de ellas

**49. ¿Cuál es la característica tecnológica principal de la segunda generación de computadores?**

- a) Los circuitos integrados
- b) Las válvulas
- c) La gran integración de los circuitos (VLSI)
- > d) Los transistores

**50. ¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?**

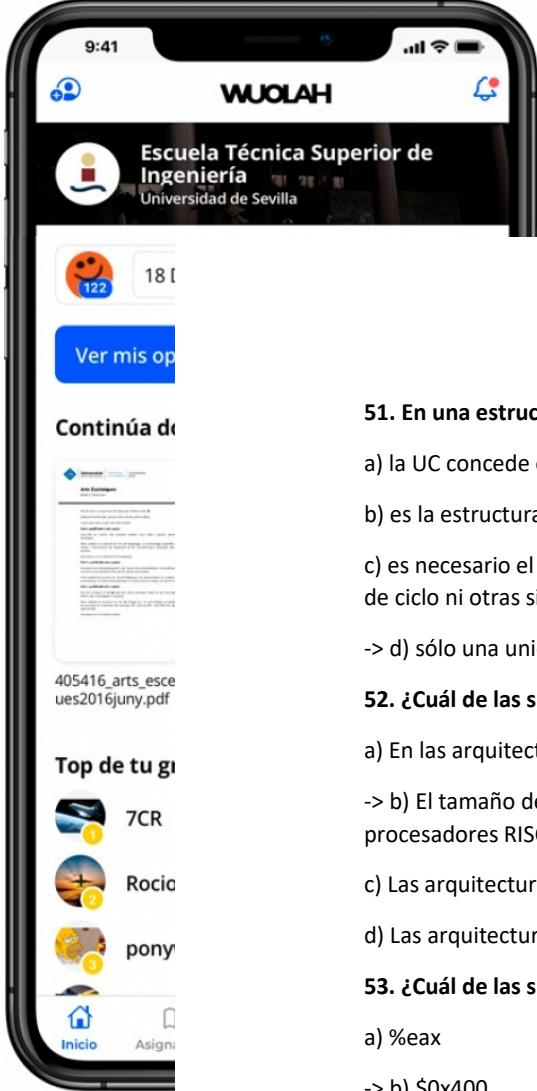
- a) posición 0: FF pos.1:FF pos.2: FF pos.3: 80

-> b) 0:80 1:FF 2:FF 3:FF

Present. tr.5,12, Intro. tr.8

- c) 0:00 1:01 2: 00 3:80

- d) Ninguna de las anteriores



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**51. En una estructura de computador de bus único (bus del sistema):**

- a) la UC concede el acceso al bus, por lo que éste funciona a la velocidad de la CPU
  - b) es la estructura más usada en los PC actuales
  - c) es necesario el arbitraje entre los maestros potenciales, no es suficiente la técnica de robo de ciclo ni otras similares.
- > d) sólo una unidad funcional puede tener el control del bus en cada momento

**52. ¿Cuál de las siguientes afirmaciones es incorrecta?**

- a) En las arquitecturas CISC hay más instrucciones que en las RISC.
- > b) El tamaño de una instrucción en lenguaje máquina siempre ocupa dos bytes en los procesadores RISC.
- c) Las arquitecturas RISC son del tipo registro-registro.
- d) Las arquitecturas RISC simplifican la decodificación.

**53. ¿Cuál de las siguientes expresiones representa un direccionamiento inmediato?**

- a) %eax
- > b) \$0x400
- c) 8(%ebp)
- d) (%eax)

**54. Si N es el número de instrucciones máquina de un programa, F es la frecuencia de reloj, y C el número promedio de ciclos por instrucción, el tiempo de ejecución del programa será:**

- a) N·F/C
- b) N·F·C
- c) N/(F·C)
- > d) N·C/F

**55. El programador de lenguaje ensamblador necesita conocer:**

- a) la microarquitectura del procesador.
- > b) la arquitectura del ordenador.
- Intro. tr.4
- c) el diseño RTL del procesador.
- d) todas las anteriores son ciertas.

**56. El primer computador electrónico basaba su funcionamiento en:**

-> a) tubos de vacío

1<sup>a</sup> generación

b) circuitos integrados LSI

3<sup>a</sup>-4<sup>a</sup> generación

c) núcleos de ferrita

2<sup>a</sup>-3<sup>a</sup> generación, tecnología RAM, no tecnología de conmutación

d) amplificadores operacionales

computadores analógicos

**57. En una arquitectura de acumulador, la instrucción LOAD X:**

a) transfiere el contenido del registro X a la memoria

no existe registro X en una máquina acumulador pura

el argumento de LOAD es una posición de memoria X

el 6502/6510 sí tenía, además de acumulador A, índices X e Y, y correspondientemente tenía instrucciones LDA, LDX, LDY

b) suma M(X) al acumulador

sería ADD X

-> c) transfiere el contenido de la posición de memoria X al acumulador

d) transfiere el contenido del acumulador a la posición de memoria X

sería STORE X

**58. Un computador con 8 bits en el bus de direcciones puede direccionar como máximo:**

a) 16384 palabras

-> b) 256 palabras

c) 8192 palabras

d) 1024 palabras

**59. ¿En qué generación, dentro de la historia de los computadores digitales, aparecieron la microprogramación, la segmentación de cauce, la memoria cache, los S.O. multiusuario y la memoria virtual?**

a) 2<sup>a</sup> generación (1955-65)

-> b) 3<sup>a</sup> generación (1965-75) //en la 3<sup>o</sup> generación se inventó casi todo.

c) 4<sup>a</sup> generación (1975-85)

d) esas innovaciones se repartieron a lo largo de varias generaciones, no sólo una

**60. En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1B), media palabra (2B) y palabra (4B), si se almacena en la posición 0xBABC una palabra de valor -2, ¿qué se obtendría al consultar la media palabra en la posición 0xBABE?**

- a) 0
- b) 1
- c) -2
- > d) -1

**61. ¿En qué registro está contenido el último dato (o instrucción) leído de memoria, o el dato que se va a escribir en memoria?**

- a) PC.
- > b) MBR.
- c) MAR.
- d) Acumulador.

**62. El bus del sistema es**

- a) en un sistema con bus único, todo el bus salvo la parte relacionada con E/S (SATA, GPU, USB, Ethernet, etc)
- b) el que conecta las distintas partes del sistema: UC, ALU, E/S, M
- > c) el que conecta CPU-M, ya sea un sistema con bus único o con múltiples buses
- d) en un sistema con buses separados, el que conecta el sistema E/S con el resto

**63. En un sistema con dos buses separados, uno para el subsistema de memoria y otro para la E/S...**

- > a) el bus que une la memoria y el procesador suele funcionar a la velocidad de la memoria
- b) el bus de E/S funciona a la velocidad del periférico más rápido
- c) ambos buses tienen que tener el mismo ancho de banda
- d) Ninguna de las respuesta anteriores es cierta

**64. El objetivo de un diseño CISC es...**

- a) disminuir la frecuencia de reloj.
- b) disminuir el número medio de ciclos por instrucción.
- c) disminuir el tamaño medio de instrucción.
- > d) disminuir el número de instrucciones a ejecutar por un programa.

**65. En un procesador de la familia 80x86 una variable de 32 bits, entera con signo, almacenada a partir de la dirección n contiene: 0xFF en la dirección n, 0xFF en la dirección n+1, 0xFF en la dirección n+2 y 0xF0 en la dirección n+3. ¿Cuánto vale dicha variable?**

a) 4294967280

b) 16

c) -16

-> d) -251658241

Como es little-endian, se trata de un número negativo de gran magnitud, y éste es el único con ese aspecto

**66. ¿Cuál de las siguientes afirmaciones es incorrecta?**

a) El formato de una instrucción nos indica el significado de cada bit de la instrucción

b) Todas las instrucciones deben tener código de operación

c) No siempre es necesario indicar la dirección de la siguiente instrucción

-> d) Todas las instrucciones deben tener operando fuente y operando destino

**67. ¿Cuál de las siguientes afirmaciones es incorrecta?**

a) El direccionamiento indexado es útil para manejo de vectores

-> b) El direccionamiento indirecto indica la dirección del operando

c) En el direccionamiento inmediato el dato se encuentra en la propia instrucción

d) En el direccionamiento implícito no se indica la ubicación del operando

No hemos explicado detalladamente el implícito, evitar esta pregunta

**68. Sea un computador con 48 registros y 200 instrucciones máquina. ¿Cuántas direcciones de memoria permite el formato de la instrucción de 32 bits hipotética "beqz r1, r2, dir"?**

**(En este enunciado el símbolo ^ representa potenciación)**

-> a)  $2^{12}$  //En la antigua ECI se hacían estos cálculos, códigos Hamming, etc. En la nueva EC no se ha explicado con tanto detalle. Evitar esta pregunta.

b)  $2^{14}$

c)  $2^{18}$

d)  $2^{16}$

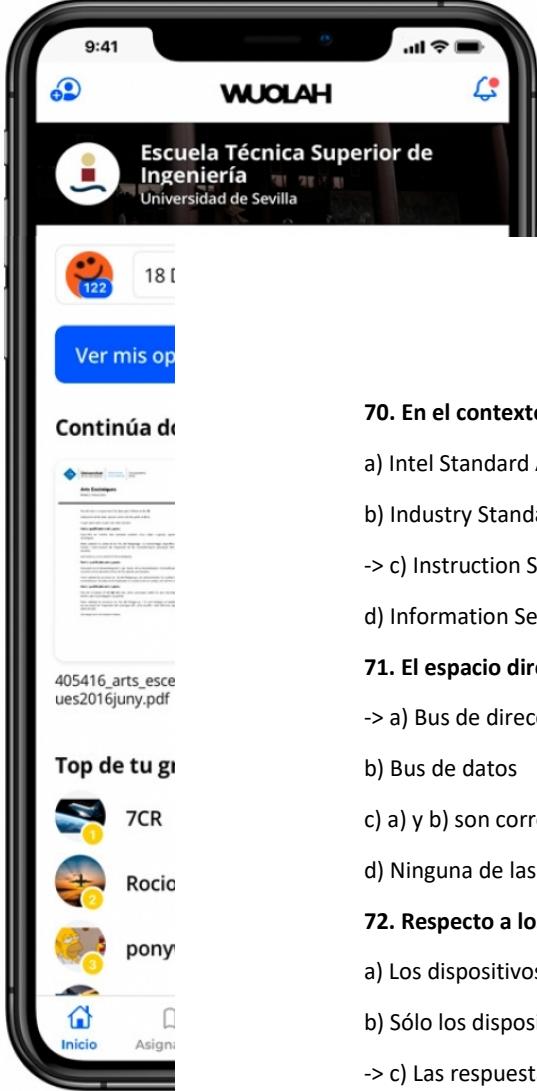
**69. En las arquitecturas RISC hay...**

-> a) muchos registros y pocos modos de direccionamiento.

b) pocos registros y muchos tipos de instrucciones.

c) pocos modos de direccionamiento y muchos formatos de instrucción.

d) pocas instrucciones muy rápidas con muchos modos de direccionamiento.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**70. En el contexto del lenguaje máquina, el acrónimo ISA suele referirse a:**

- a) Intel Standard Architecture
- b) Industry Standard Architecture
- > c) Instruction Set Architecture
- d) Information Security Architecture

**71. El espacio direccionable de memoria de un computador depende del diseño del:**

- > a) Bus de direcciones
- b) Bus de datos
- c) a) y b) son correctas
- d) Ninguna de las anteriores es correcta

**72. Respecto a los dispositivos activos y pasivos en un bus podemos decir que:**

- a) Los dispositivos pasivos sólo pueden convertirse en esclavos
- b) Sólo los dispositivos activos pueden convertirse en maestros
- > c) Las respuestas a y b son ciertas
- d) Las respuestas a y b son falsas

**73. En una memoria de bytes que contuviera a partir de la posición 0 los valores 1,0,0,0,0xFE,0xFF,0xFF,0xFF, se puede decir que...**

- a) Hay una palabra de 16bit big-endian con valor 1 en la posición 0
- b) Hay una palabra de 16bit little-endian con valor 254 en la posición 3
- c) Hay una palabra de 32bit little-endian con valor -1 en la posición 4
- > d) Todas las respuestas anteriores son incorrectas

**74. En la ejecución de una instrucción...**

- a) la UC activa las señales de control que envía por el bus de direcciones
- b) siempre se altera el registro de estado
- > c) la ALU realiza las operaciones aritméticas y lógicas
- d) el registro de instrucción se va incrementando para apuntar a la siguiente instrucción

**75. ¿En qué generación, dentro de la historia de los computadores digitales, aparece la microprogramación?**

- a) primera
- b) segunda
- > c) tercera
- d) cuarta

**76. ¿En qué generación, dentro de la historia de los computadores digitales, aparece la segmentación de cauce?**

- a) primera
- b) segunda
- > c) tercera
- d) cuarta

**77. La ecuación básica de rendimiento calcula**

- > a) cuánto tiempo tarda en ejecutarse un programa concreto conociendo su número de instrucciones y el número de etapas (promedio) y la frecuencia del procesador
- b) el promedio de las ganancias obtenidas con una serie de programas de punto entero
- c) la media geométrica de los cocientes entre los tiempos de ejecución de una serie de programas predeterminados
- d) cómo de mejor es un procesador frente a otro, conociendo las prestaciones de las respectivas UC, ALU, E/S y M

**78. Si usamos una estructura de bus con DMA:**

- > a) la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.
- b) al bus del sistema sólo se conecta la CPU y la MP, ya que el DMA se conecta directamente a MP para realizar las transferencias de datos.
- c) la velocidad de este controlador establece la velocidad del bus del sistema.
- d) podemos prescindir de controladores de E/S ya que el controlador de DMA se ocupa de controlar las transferencias hacia/desde los periféricos.

**79. ¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop?:**

**push #1**

**push #2**

**push #3 ; pop a**

**push #4**

**pop a**

**pop a**

- > a) 1

- b) 1 y 2

- c) 1, 2, 3 y 4

- d) 10

**80. El registro MBR...**

- a) especifica la dirección en memoria de la palabra que va a ser escrita o leída
- > b) contiene el valor que va a ser almacenado en la memoria, o bien se usa para recibir un valor procedente de la memoria
- c) contiene la dirección de la próxima instrucción que va a ser captada de memoria
- d) contiene el código de operación de la instrucción que se está ejecutando

**81. En una arquitectura de registros de propósito general (a nivel de lenguaje máquina):**

- a) operar usando registros es más rápido.
- b) la generación de código resulta más simple que en arquitecturas de pila o acumulador.
- c) se evita el cuello de botella (por ejemplo, pila, o acumulador) que otras arquitecturas presentan al evaluar expresiones aritméticas complejas
- > d) todas las respuestas anteriores son ciertas.

**82. En la arquitectura Von Neumann...**

- a) los bloques principales son la unidad de control, la ALU y la CPU.
- > b) el programa se encuentra residente en memoria.
- c) los registros se encuentran en la memoria principal.
- d) Todas son ciertas.

**83. Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos y un computador Y tarda 14 segundos en ejecutar ese mismo programa, ¿cuántas veces es más rápido el computador Y que el X?**

- a) 1,538
- > b) 1,857
- c) 0,538
- d) 12

**84. Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro**

- a) Contador de programa
- b) De direcciones
- c) Puntero de pila
- > d) De instrucción

**85. No en todas las instrucciones máquina hay una fase de**

- a) decodificación
- > b) captura de operandos
- c) captación
- d) ejecución

**86. El conjunto de todos los atributos de un sistema que son visibles para el programador y son necesarios para programar en lenguaje máquina se denomina:**

- a) repertorio de instrucciones máquina
- > b) arquitectura del computador

Intro. tr.4

- c) conjunto de componentes físicos del computador
- d) organización del computador

**87. Si en un bus de direcciones de 32 bits se decodifica parcialmente la dirección de un dispositivo de 32 posiciones usando 22 bits, ¿cuántas veces aparecerá repetido en el mapa de memoria?**

- a) 10
- b) 1024
- > c) 32

el dispositivo tiene 32 puertos, usa 5 bits de direccionamiento (probablemente los 5 LSB), pero en lugar de decodificarse con 27, se usan sólo 22 bits (probablemente los 22 MSB) de manera que quedan 5 bits (probablemente intermedios) sin usar, que pueden tomar 32 combinaciones posibles

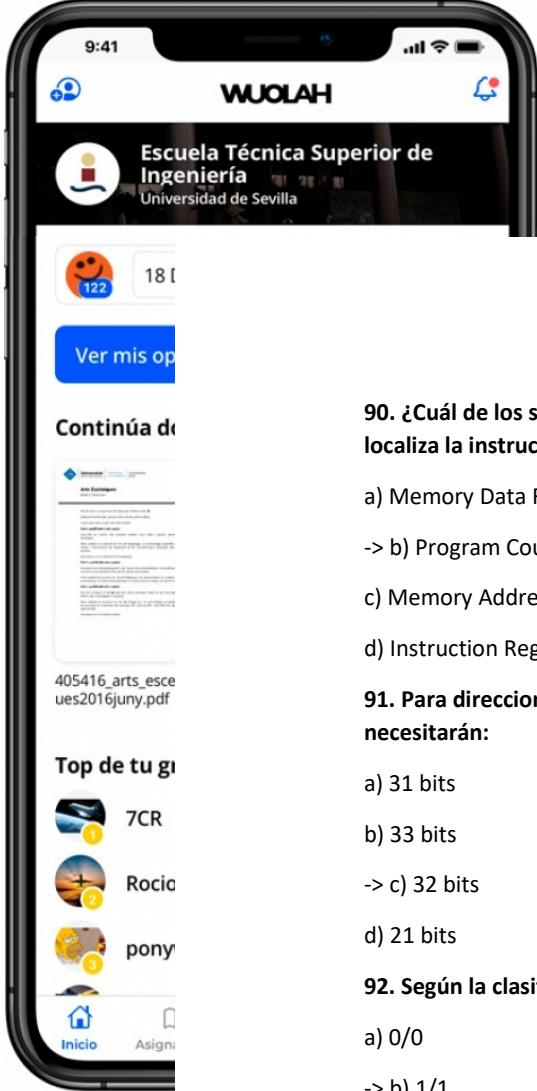
- d) 16

**88. ¿Cuál de las siguientes afirmaciones es falsa?**

- a) el bus de control transporta señales de estado
- b) el bus de datos es bidireccional
- c) el bus de direcciones es unidireccional
- > d) la anchura del bus de datos es de 16 bits

**89. Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "big-endian", quedará almacenada a partir de la posición 0x1000 como:**

- a) M[0x1000]=0xA6 y M[0x1001]=0x91
- > b) M[0x1000]=0x89 y M[0x1001]=0x65
- c) M[0x1000]=0x91 y M[0x1001]=0xA6
- d) M[0x1000]=0x65 y M[0x1001]=0x89



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**90. ¿Cuál de los siguientes registros se utiliza para guardar la dirección de memoria donde se localiza la instrucción siguiente?**

- a) Memory Data Register
- > b) Program Counter
- c) Memory Address Register
- d) Instruction Register

**91. Para direccionar una memoria de bytes en la que quepan 1G palabras de 32 bits se necesitarán:**

- a) 31 bits
- b) 33 bits
- > c) 32 bits
- d) 21 bits

**92. Según la clasificación m/n, las máquinas de acumulador son de tipo**

- a) 0/0
- > b) 1/1
- c) 2/2 ó 2/3
- d) 1/2

**93. En la captación de un operando que reside en memoria:**

- a) en MBR indicamos la dirección donde está y en MAR lo recogemos
- b) en MBR indicamos la dirección donde está y en IR lo recogemos
- c) en MAR indicamos la dirección donde está y en IR lo recogemos
- > d) en MAR indicamos la dirección donde está y en MBR lo recogemos

**94. En el arbitraje de un bus...**

- a) los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
- > b) si hay un único dispositivo pasivo, siempre funciona como esclavo
- c) si hay varios dispositivos activos, siempre funcionan como maestros
- d) todas las respuestas anteriores son ciertas

**95. ¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:**

**push #4**

**push #7**

**add**

**push #10**

**sub**

-> a) 1

b) 11, 1

c) 4, 11, 1

d) 4, 7, 10

**96. Se dice que las máquinas con arquitectura Von Neumann siguen un modelo de programa...**

a) Cableado.

b) Microprogramado.

c) Externo.

-> d) Almacenado.

**97. Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria. Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de  $2^{10}$ , ¿qué tamaño de memoria admite el computador?**

a)  $2^{10}$  palabras

-> b)  $2^{12}$  palabras

c)  $2^{13}$  palabras

d) Ninguna de las anteriores

**98. ¿De qué tipo son los procesadores Intel que usamos en los laboratorios?**

a) puede ajustarse mediante un bit de control en el registro CR0 que funcionen como little- o big-endian

b) el concepto de endian no es aplicable a estas máquinas, ya que un registro del procesador no cabe en una posición de memoria

c) big-endian

-> d) little-endian

**99. ¿De qué depende el tamaño del contador de programa?**

- a) de la longitud del código de operación
- b) del ancho del bus de datos
- c) el tamaño no importa

-> d) ninguna de las anteriores es cierta

**100. El instrumento GIADA de la sonda espacial ROSETTA (diseñado en Granada) está basado en un microprocesador 8086 y el siguiente mapa de memoria:**

**RAM volátil: 00000 - 0FFFF**

**RAM no volátil: 10000 - 1FFFF**

**ROM: F0000 - FFFFF**

**¿Cuál es el tamaño total de la memoria?**

- a) 3MB
- b) 48KB
- c) 2MB

-> d) 192KB

**101. Un computador que utilice el sistema big-endian, almacena el número 0x2143 a partir de la dirección 0 como:**

- a) M[0]=0x43 y M[1]=0x21
- > b) M[0]=0x21 y M[1]=0x43
- c) M[0]=0x12 y M[1]=0x34
- d) M[0]=0x34 y M[1]=0x12

**102. El direccionamiento relativo a registro base utiliza...**

- a) dos desplazamientos contenidos en la propia instrucción.
- b) un registro y un factor de escala
- c) dos registros.

-> d) un registro y un desplazamiento

**103. El computador EDVAC, propuesto por John Von Neumann, presentaba dos importantes diferencias respecto al ENIAC:**

- > a) Empleaba aritmética binaria y permitía trabajar con un programa almacenado.
- b) Se programaba enchufando centenares de clavijas y empleaba aritmética octal.
- c) Era electromecánico y de propósito específico.
- d) Utilizaba transistores y memoria de semiconductor.

**104. ¿Cuál de las siguientes características es posterior a la segunda generación de computadores?**

- a) Lenguaje ensamblador.
- b) Transistor.
- > c) RISC.
- d) Memoria de núcleos de ferrita.

**105. ¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?**

- a) registro-registro
- b) registro-memoria
- > c) memoria-memoria
- d) ninguna de las anteriores es cierta

**106. En las últimas generaciones de computadores la mejora de prestaciones viene dada por:**

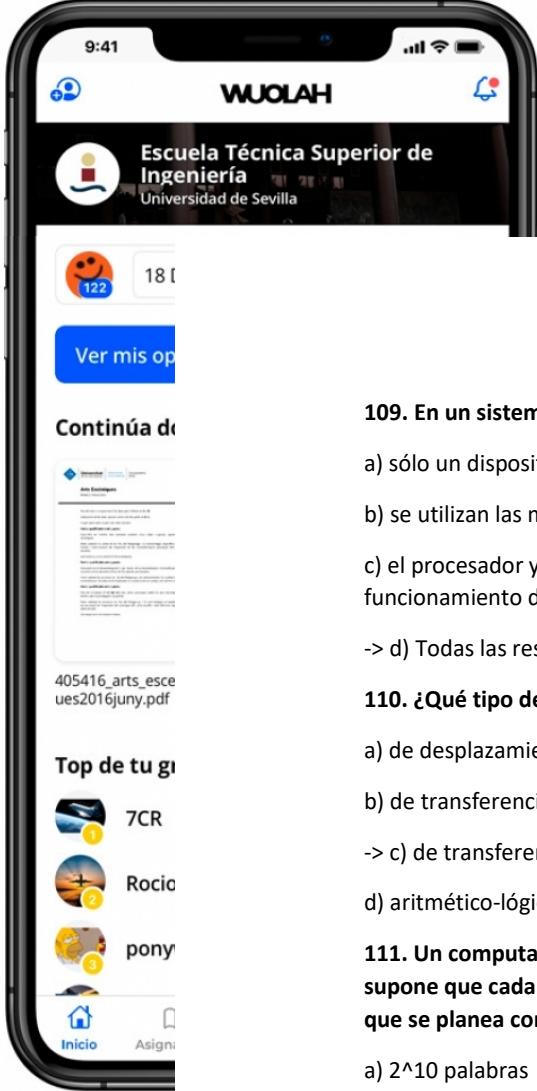
- a) avances en las tecnologías de fabricación.
- > b) avances en tecnología y avances en la estructura y arquitectura del computador.
- c) avances en los sistemas operativos y aplicaciones.
- d) avances en la estructura y arquitectura del computador.

**107. Una dirección de memoria se refiere siempre a:**

- a) una palabra
- b) 16 bits
- c) un byte
- > d) ninguna de las anteriores

**108. Según la clasificación m/n, las máquinas con arquitectura R/R son de tipo**

- a) x/x con x=2,3
- b) x/0 con x=2,3
- > c) 0/x con x=2,3
- d) 0/0



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## 109. En un sistema con un único bus...

- a) sólo un dispositivo puede escribir en un instante dado en el bus
- b) se utilizan las mismas líneas de control para conectar todos los dispositivos
- c) el procesador y los periféricos pueden funcionar a diferentes velocidades si el funcionamiento del bus es asíncrono
- > d) Todas las respuestas anteriores son ciertas

## 110. ¿Qué tipo de instrucciones se emplean más en una arquitectura de acumulador?

- a) de desplazamiento y rotación
- b) de transferencia de datos entre registros
- > c) de transferencia de datos con memoria
- d) aritmético-lógicas

## 111. Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria. Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de $2^{10}$ , ¿qué tamaño de memoria admite el computador?

- a)  $2^{10}$  palabras
- > b)  $2^{12}$  palabras
- c)  $2^{13}$  palabras
- d) Ninguna de las anteriores

## 112. ¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?

- a) posición 0: FF pos.1:FF pos.2: FF pos.3: 00
- b) 0:00 1:FF 2:FF 3:FF
- c) 0:00 1:01 2: 00 3:80
- > d) Ninguna de las anteriores

## 113. Una memoria que está estructurada en palabras de 8 bits tiene una capacidad de 64 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?

- a) 24
- b) 12
- > c) 13
- d) 8

**114. El ancho de palabra de una memoria corresponde a:**

- a) La longitud del registro de direcciones de la memoria.
- b) El número de posiciones que la componen.
- c) El número que identifica únicamente cada posición de la memoria.
- > d) La longitud del registro de datos de la memoria.

**115. ¿Cuál de las siguientes afirmaciones acerca de la arquitectura Von Neumann es cierta?**

- > a) La separación entre almacenamiento y unidad de procesamiento es una de las ideas contempladas en la arquitectura Von Neumann.
- b) La arquitectura Von Neumann es un diseño que sitúa el programa en un almacenamiento distinto al de los datos.
- c) Existe un consenso general en considerar justo el término "arquitectura von Neumann", ya que las ideas de esta arquitectura fueron completamente originales de John Von Neumann y no influenciadas por sus colaboradores o predecesores.
- d) Para ejecutar un programa en una máquina Von Neumann, es necesario volver a cablear o incluso rediseñar la máquina.

# TEST EC TEMA 2

**1. Una instrucción de "salto si menor", para números sin signo, tiene que comprobar el valor de:**

- a) el bit de cero
- b) los bits de signo y desbordamiento
- > c) el bit de acarreo
- d) el bit de signo

**2. ¿Qué valor contendrá el registro edx tras ejecutar las dos instrucciones siguientes?**

```
movl $-1, %edx
movb $1, %dl
```

Usuario Profesores

- a) 00000000 00000000 00000000 00000001
- b) 11111111 11111111 11111111 11111111
- c) 00000001 00000000 00000000 00000000
- > d) 11111111 11111111 11111111 00000001

**3. En los modos de direccionamiento del tipo Desplazamiento(Base,Indice,Factor Escala), puede usarse como**

- a) base, cualquiera de los 8 registros enteros salvo %esp
- b) factor de escala, cualquier constante de 1, 2, 4 u 8 bytes
- > c) desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)
- d) índice, también cualquiera salvo %ebp

**4. Respecto a requisitos de alineamiento de structs en gcc/IA32 x86 y x86-64, alguna de las siguientes afirmaciones es falsa**

- a) en x86-64 Linux alinea double a 8x (Windows también)
- b) en x86 Linux alinea double a 4x (Windows no)
- > c) en x86-64 Linux alinea float a 8x (Windows también)

Transparencias corregidas, ver Tema 2.5 tr.6

- d) en x86 Linux alinea long double a 4x (Windows también)

**5. La primera instrucción ensamblador de una subrutina compilada con gcc en Linux/x86 cdecl suele ser:**

- a) push %ebx
- b) mov %esp, %ebp
- c) pop %ebx
- > d) push %ebp

**6. La instrucción JGE / JNL provoca un salto si...**

- a) CF = 1

sería jc/jb

- > b) OF = SF

basta recordar que "Less" no era un flag solo (es OF^SF)

recordar también que "Below" comprueba CF

- c) SF = 1

sería js

- d) SF = 0

sería jns

**7. El lenguaje máquina es...**

- > a) difícil de codificar manualmente.
- b) portable entre arquitecturas.
- c) una alternativa razonable al uso del lenguaje ensamblador.
- d) fácilmente legible por el programador.

**8. La arquitectura x86-64 tiene:**

- a) 8 registros de propósito general (RPG) de 64 bits (%rax, %rbx, ... %rsp, %rbp)
- b) 64 registros RPG de 64 bits
- > c) 16 registros RPG de 64 bits
- d) 32 registros RPG de 64 bits

## **9. El marco de pila en x86-64 Linux...**

- a) se crea para funciones en las que GCC no puede evitar que RBP baje más, como por ejemplo: que haya demasiadas variables locales (y no quepan en registros), o que haya que salvar algún registro salva-invocado
- b) no existe, porque RBP no es registro especial en x86-64
- > c) se crea para funciones en las que GCC no puede evitar que RSP baje más, como por ejemplo: que haya que calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función
- d) sólo se crea para funciones que invocan anidadamente a otra función (procedimientos padre, no hojas)

## **10. La instrucción leave equivale a:**

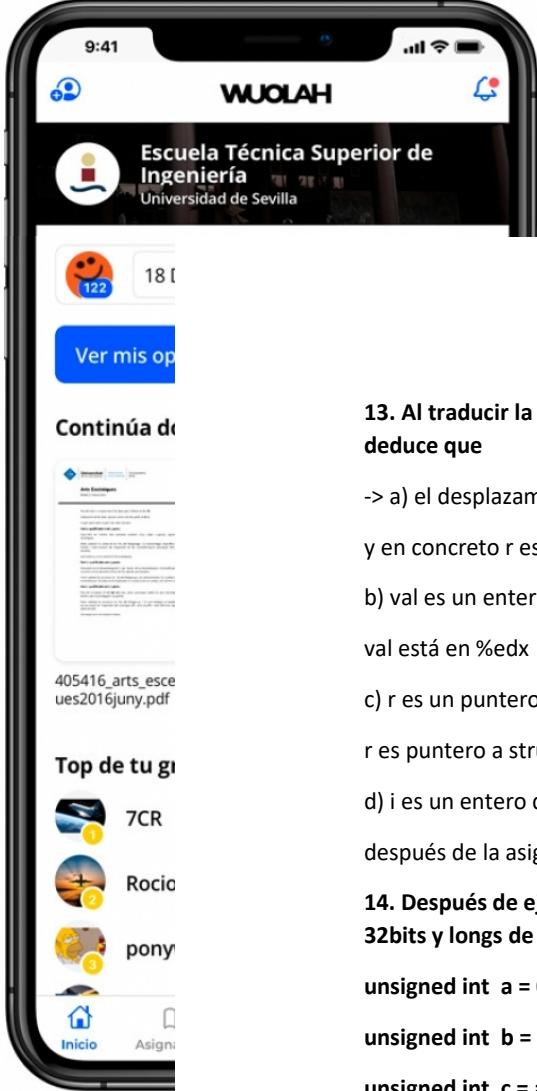
- > a) movl %ebp, %esp; popl %ebp
- b) pushl %ebp; movl %esp, %ebp
- c) movl %esp, %ebp; popl %esp
- d) pushl %esp; movl %ebp, %esp

## **11. El comienzo de un procedimiento que siga la convención cdecl es:**

- a) mov %esp,%ebp; push %ebp
- b) push %ebp; mov %ebp,%esp
- c) mov %ebp,%esp; push %ebp
- > d) push %ebp; mov %esp,%ebp

## **12. Las instrucciones que asignan la dirección 0x78e00 a un puntero situado en la posición 0xff00 son:**

- a) mov 0x78e00,%eax  
lea %eax,0xff00
- b) mov \$0x78e00,%eax  
lea %eax,0xff00
- c) mov 0x78e00,%eax  
mov %eax,0xff00
- > d) mov \$0x78e00,%eax  
mov %eax,0xff00



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

**13. Al traducir la sentencia C r->i = val; gcc genera el código ASM movl %edx, 12(%eax). Se deduce que**

- > a) el desplazamiento de i en \*r es 12
- y en concreto r está en %eax
- b) val es un entero que vale 12
- val está en %edx
- c) r es un puntero que apunta a la posición de memoria 12

r es puntero a struct y seguramente valdrá 0x0804XXXX en ejecutables x86

- d) i es un entero que vale 12

después de la asignación, i vale lo que había en %edx (que es val)

**14. Despues de ejecutar el siguiente código, ¿qué variables serán igual a 0? (Suponer ints de 32bits y longs de 64bits)**

```
unsigned int a = 0xffffffff;
unsigned int b = 1;
unsigned int c = a + b;
unsigned long d = a + b;
unsigned long e = (unsigned long)a + b;
```

- a) Ninguna

- b) c

- > c) c y d

En el problema 3.4 sólo se explica que una extensión de tamaño se hace según el fuente sea signed (extensión sgn) o unsigned (ext. con ceros), pero no se explica la sección 2.2.6 (y 2.2.5) en donde se aclara que las operaciones que impliquen a algún unsigned se hacen en unsigned. Evitar esta pregunta en el futuro.

- d) c, d, y e

**15. ¿Cuál de las siguientes expresiones toma el valor 0x01 si x es múltiplo de 32 y 0x0 en caso contrario? Asumir que x es unsigned int.**

- a) (x & 0x1f)

- > b) !(x & 0x1f)

- c) (x | 0x3f)

- d) !(x & 0x3f)

**16. ¿Cuál de las siguientes afirmaciones es correcta?**

- a) El lenguaje máquina es igual para todos los computadores.
- > b) Las instrucciones en lenguaje máquina se almacenan y tratan como cadenas de unos y ceros.
- c) El lenguaje ensamblador es igual para todos los computadores.
- d) Las instrucciones en lenguaje ensamblador se almacenan y tratan como cadenas de unos y ceros.

**17. Al ejecutar el fragmento de código:**

```
leal -48(%eax), %edx
cmpl $9, %edx
ja .L2
```

**se salta a .L2 si el contenido del registro %eax:**

- a) es mayor o igual que 57
- b) está dentro del intervalo [48,57]
- > c) está fuera del intervalo [48,57]
- d) es mayor o igual que 48

**18. De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...**

- > a) El bucle do-while
- b) El bucle for
- c) La selección switch-case
- d) El bucle while

**19. Respecto a registros de propósito general (RPG), el 80386 tiene:**

- a) 8 registros de 8 bits
- > b) 8 registros de 32 bits
- Tema1 tr.26
- c) 16 registros de 64 bits
- d) 16 registros de 16 bits

**20. La instrucción seta %al (seta significa "set if above"):**

- a) Pone AL a 1 si CF=1 y ZF=0
- > b) Pone AL a 1 si CF=0 y ZF=0
- c) Pone AL a 1 si CF=1 o ZF=1
- d) Pone AL a 1 si CF=0 o ZF=0

**21. Se ha declarado en un programa C la variable int val[5]={1,5,2,1,3} . ¿Cuál de las tres primeras opciones (a, b, o c) es FALSA?**

- > a) val[1] == 1
- b) &val[3] == val+3
- c) sizeof(val) == 20
- d) No se puede marcar ninguna de ellas, todas (a, b y c) son ciertas

**22. Respecto a registros base e índice en IA32, la excepción es que**

- a) EBP no puede ser registro índice
- b) ESP no puede ser registro base
- c) EBP no puede ser registro base
- > d) ESP no puede ser registro índice

**23. ¿Cuál de las siguientes instrucciones máquina copia en EAX la dirección efectiva resultante de la operación EDX\*4 + EBX?**

- a) movl 4(%edx, %edx), %eax
- b) leal 4(%edx, %edx), %eax
- c) movl (%ebx, %edx, 4), %eax
- > d) leal (%ebx, %edx, 4), %eax

**24. Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/IA32 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como:**

- > a) -k(%ebp), siendo k un número constante positivo relativamente pequeño
- b) k(%esp), siendo k un número constante positivo relativamente pequeño
- c) var (el nombre de la variable representa su posición de memoria)
- d) k(%ebp), siendo k un número constante positivo relativamente pequeño

**25. Por x86-64 se entiende la misma arquitectura de repertorio (ISA) que**

- a) IA32
- > b) AMD64
- c) x86
- d) IA64

**26. Cuál de las instrucciones máquina siguientes es incorrecta en x86-64:**

- a) movl (%rdi,%rcx,4), %edx
- b) testl %edx, %edx
- c) addq \$1, %rcx
- > d) movl %r8, %eax

**27. Usando el repertorio IA32, para intercambiar el valor de 2 variables (por ejemplo A: .int 1 y B: .int 2) se pueden usar...**

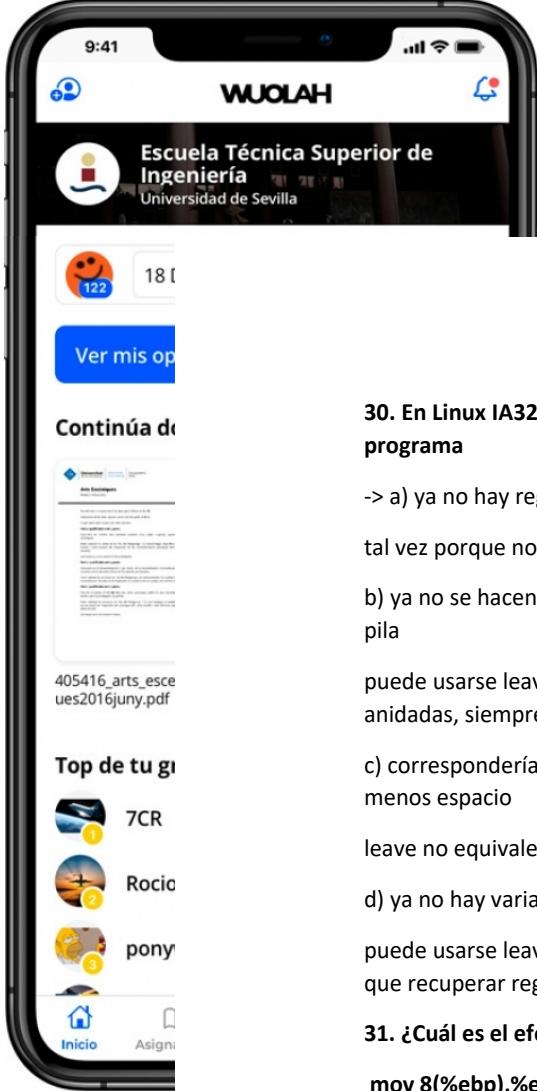
- a) una instrucción mov y una instrucción lea
- > b) 4 mov, no menos (debido a la arquitectura R/M) //ver la función swap()
- c) dos instrucciones mov
- d) 3 mov, no menos (se le llama "intercambio circular")

si fuera arquitectura M/M se podría hacer intercambio circular:

```
mov A,R1
mov B,A; // esta instrucción no es de repertorio R/M
mov R1,B
```

**29. Si EBX=0x00002a00 y EDI=0x0000000a, ¿cuál es la dirección efectiva en la instrucción add 0x64(%ebx,%edi,2),%eax?**

- a) 0x00005478
- b) 0x000054dc
- c) 0x00002a70
- > d) 0x00002a78



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**30. En Linux IA32, si gcc usa la instrucción leave se puede asegurar que en ese punto del programa**

- > a) ya no hay registros salva-invocado que recuperar tal vez porque no había ninguno, para empezar
- b) ya no se hacen llamadas anidadas y por tanto no hay parámetros que ocupen espacio en pila

puede usarse leave habiendo espacio reservado en pila para argumentos de llamadas anidadas, siempre que no haya que recuperar registros salva-invocado

- c) correspondería emitir la secuencia de salida pop/ret, pero leave hace lo mismo y ocupa menos espacio

leave no equivale a ret, y sólo hace pop %ebp

- d) ya no hay variables locales que destruir

puede usarse leave para destruir rápidamente todas las variables locales, siempre que no haya que recuperar registros salva-invocado

**31. ¿Cuál es el efecto de la instrucción siguiente?**

**mov 8(%ebp),%ecx**

- > a) Suma 8 al contenido de ebp, trata la suma como una dirección de memoria y almacena el contenido de esa dirección en ecx
- b) Suma los contenidos de ebp y de la dirección de memoria 8 y almacena la suma en ecx
- c) Suma 8 al contenido de ebp y almacena la suma en ecx
- d) Suma 8 al contenido de la posición de memoria cuya dirección está almacenada en ebp, y almacena la suma en ecx

**32. En el direccionamiento indirecto a través de registro, la dirección efectiva...**

- a) se encuentra en el registro de instrucción.
- b) se calcula como la suma del contenido de dos registros.
- c) se encuentra en una dirección de memoria.
- > d) se encuentra en un registro general del procesador.

**33. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?**

- > a) EBX
- b) EAX
- c) EDX
- d) ECX

**34. Las instrucciones JB y JNAE del Pentium provocan un salto si...**

- a) SF == 1
- b) ZF != SF
- > c) CF == 1

below es la condición más fácil: sin signo 3-4 = debo una

- d) ZF == 0

**35. ¿En qué registro se pasa el primer argumento a una función en Linux gcc x86-64?**

- a) edx
- > b) edi
- c) ecx
- d) esi

**36. La convención de llamada Linux/GCC x86-32 considera, respecto a convenios de uso de registros:**

- a) 8 registros salva-invocante, 6 registros salva-invocado, y 2 especiales
- b) Algunos registros para pasar argumentos, otros salva-invocante, otros salva-invocado, dos especiales
- > c) 3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales
- d) Algunos registros salva-invocante, otros salva-invocado, uno especial

**37. Con el repertorio IA32, para sumar %eax y %ebx dejando el resultado en %ecx se podría hacer lo siguiente:**

- > a) lea (%eax, %ebx, 1), %ecx
- b) lea %ecx, %ebx, %eax
- c) lea %eax, %ebx, %ecx
- d) lea %ecx, [%eax, %ebx]

**38. Suponiendo que todos los registros inicialmente contienen el valor 1, ¿cuál es el valor de ECX tras la ejecución de la siguiente secuencia de instrucciones?**

```
mov $4, %eax
mov $3, %ebx
lea (%eax, %eax), %ecx
sub %ebx, %ecx
imul %ecx, %eax
```

- a) 6
- b) 36
- > c) 5
- d) 20

**39. Diferencias gcc Linux IA32/x86-64: marcar la respuesta falsa**

- > a) el tipo double pasa de 4 B a 8 B
- siempre 8 B
- b) los enteros largos (long) pasan de 32 a 64 bits
- c) long double pasa de 10/12 B a 16 B
- d) los punteros (void\*) pasan de 32 a 64 bits

**40. Si AX = 0xFA50 y ejecutamos XOR \$0xFF, AX**

- a) El registro AL se pone a 0.
- b) El registro AH se pone a 0.
- > c) Se realiza el complemento a 1 de AL.
- d) Se realiza el complemento a 1 de AH.

**41. La instrucción movzbl %al, %eax**

- > a) Copia en %eax el valor sin signo almacenado en %al, rellenando con ceros

Tema 2.1 tr. 28, Hallaron, p.205-206

- b) Copia en %eax el valor de %al si el indicador de cero está activado
- c) Copia en %eax el valor del indicador de cero
- d) Pone a 0 el registro %eax

**42. En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?**

-> a) mov \$-1, %edi

mov no afecta a los flags

b) mov \$0, %edi

sub \$1, %edi

sub afecta

c) sub %edi, %edi

adc \$0xFFFFFFFF, %edi

adc afecta

d) mov \$-1, %edi

add \$0, %edi

add afecta

**43. Si la estructura struct a ocupa un espacio de 28 bytes en memoria, ¿cuántos bytes ocupa la siguiente estructura struct b cuando se compila en 64 bits?**

```
struct b {
```

```
 struct a a1;
```

```
 int i;
```

```
 struct a a2;
```

```
};
```

-> a) 60 bytes //28 + 4 + 28 = 60

b) 84 bytes

c) 24 byte // imposible: una sola struct a ya ocupa más

d) 64 bytes

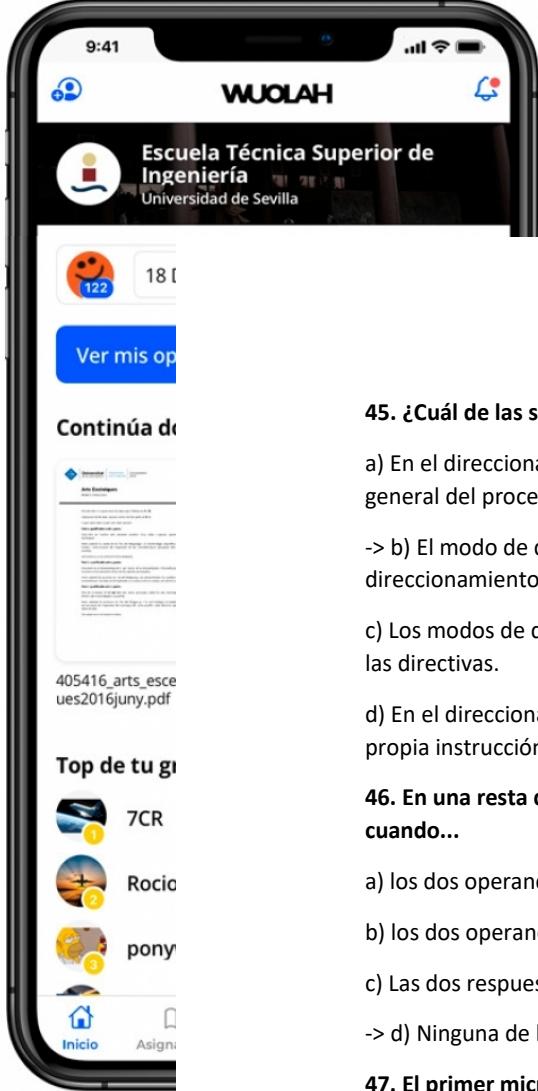
**44. ¿Cuál de las siguientes afirmaciones es falsa?**

a) Las subrutinas necesitan ser capaces de reservar espacio en memoria para las variables locales sin sobrescribir ningún dato usado por el programa que hace la llamada

b) Las subrutinas necesitan algún modo de saber desde dónde han sido llamadas para poder volver al programa que realizó la llamada cuando se completa la subrutina

c) Los programas necesitan una forma de pasar parámetros a las subrutinas y de recibir las salidas de vuelta

-> d) Las subrutinas necesitan recibir parámetros desde el programa que hace la llamada que indiquen qué registros pueden alterar y cuáles no



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## 45. ¿Cuál de las siguientes afirmaciones es cierta?

- a) En el direccionamiento inmediato, la dirección efectiva se encuentra en un registro de uso general del procesador.
- > b) El modo de direccionamiento indirecto a través de registro es el modo más sencillo de direccionamiento a memoria.
- c) Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.
- d) En el direccionamiento a registro, el objeto direccionado es una constante contenida en la propia instrucción.

## 46. En una resta de dos números en complemento a dos, se produce desbordamiento cuando...

- a) los dos operandos son negativos y el resultado es positivo.
- b) los dos operandos son positivos y el resultado es negativo.
- c) Las dos respuestas a y b son correctas.
- > d) Ninguna de las anteriores es correcta.

## 47. El primer microprocesador de 32 bits de la familia x86 fue el:

- a) 80486
- b) 80286
- c) Pentium
- > d) 80386

## 48. ¿Cuál es la diferencia entre los desplazamientos a la derecha lógico y aritmético?

- a) El aritmético inserta en el bit más a la derecha una copia del bit de signo
- b) Ninguna, la diferencia es entre los desplazamientos a la izquierda
- c) El lógico inserta siempre ceros en el bit más a la derecha
- > d) Insertan de forma distinta el bit más a la izquierda

## 49. ¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso? (salva-invocante, invocado, etc)

- a) RAX, RBX, RCX, RDX //RBX salva-invocante
  - > b) CL, DX, R8d, R9 //todos "amarillos" (args. salva-invocante) en tr.2.4.5
  - c) RBX, RSI, RDI //RBX salva-invocante
  - d) RSP, RBP
- RSP especial (el único especial)

**50. Una sentencia en C del tipo "while (test) body;" puede transformarse en código "goto" como:**

a) loop:

body;

if (test) goto loop;

-> b) if (!test) goto done;

loop:

body;

if (test) goto loop;

done:

c) loop:

if (test) goto done;

body;

goto loop;

done:

d) if (test) goto true;

goto done;

true:

body;

done:

**51. Indicar cuál es la dirección de la instrucción mov en el siguiente desensamblado, donde se ha borrado parte de la dirección**

0804xxxx: 74 12                  je 08048391

0804xxxx: b8 00 00 00 00 mov \$0, %eax

a) 08048391 + 12 = 08048403

-> b) 0804837f

Tema 2.3, tr.7-8

c) 08048391 – 12 = 08048379

sí, es esa resta, pero está mal hecha, como si fueran números en decimal

d) 0804837d

**52. El resultado de desplazar 0xFF0F a la derecha aritméticamente 4 veces es:**

- > a) 0xFFFF0
- b) 0x0FF0
- c) 0xFOFF
- d) 0xFOFO

**53. ¿Cuál es el resultado de evaluar la expresión 0b1110 ^ 0b1010 en lenguaje C?**

**(0b indica que el número está expresado en binario)**

- a) 0b1010
- > b) 0b0100
- c) 0b1111
- d) 0b0110

**54. Para crear espacio en la pila para variables locales sin inicializar suele realizarse la siguiente operación:**

- > a) Restar una cantidad positiva a ESP
- b) Restar una cantidad positiva a EBP
- c) Sumar una cantidad positiva a ESP
- d) Sumar una cantidad positiva a EBP

**55. Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?**

- a) ESP no se puede usar como registro índice
- b) El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
- c) El factor de escala S puede ser 1, 2, 4, 8
- > d) EBP no se puede usar como registro base

**56. ¿Cuál de las siguientes afirmaciones sobre compilación C->ASM es falsa?**

- > a) Puede que el compilador altere por optimización el orden de los parámetros pasados en una llamada a función, si el marco de pila resultante es más eficiente
- b) Puede que el compilador elimine por optimización construcciones C enteras (como un bucle for completo), si se conoce el resultado en tiempo de compilación
- c) Puede suceder que varias sentencias C se traduzcan por una única instrucción ASM
- d) Puede que el compilador altere el orden del código C, apareciendo antes la traducción de sentencias C posteriores

**57. El ajuste de marco de pila que gcc (Linux/IA32) prepara para todas las funciones consiste en las instrucciones**

- > a) pushl %ebp  
movl %esp, %ebp
- b) pushl %esp  
movl %ebp, %esp
- c) movl %esp, %ebp  
pushl %esp
- d) movl %ebp, %esp  
pushl %ebp

**58. ¿Cuál es la diferencia entre las instrucciones mov y lea?**

- a) mov puede usarse para copiar un registro a otro, mientras que lea no  
lea (%ebx), %eax
- b) lea puede usarse para copiar un registro a otro, mientras que mov no  
mov %ebx, %eax
- c) lea accede a la posición indicada, mientras que mov no lo hace
- > d) mov accede a la posición indicada, mientras que lea no lo hace

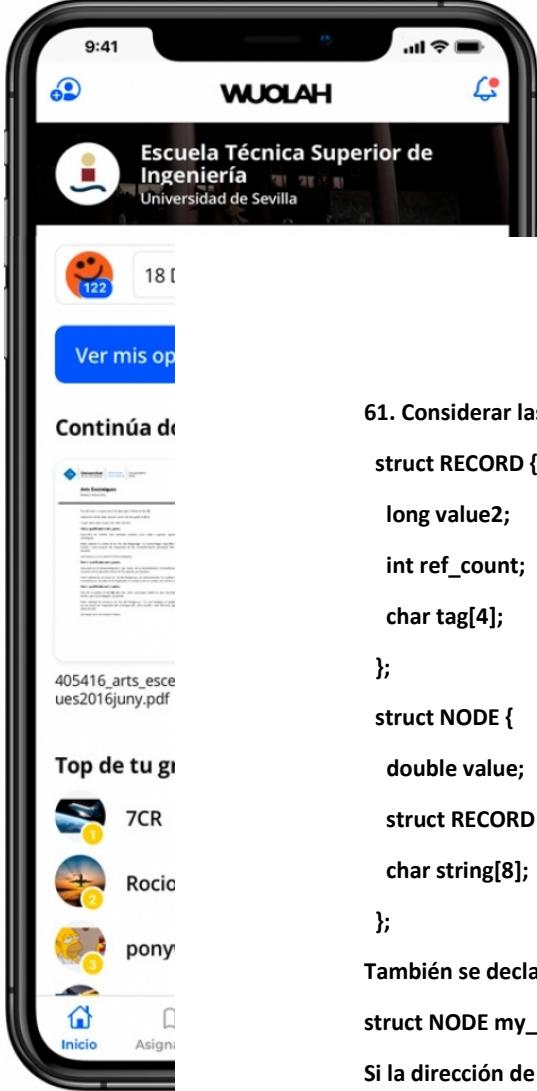
**59. Si la variable val está almacenada en ebx y la variable x está almacenada en eax, la sentencia val ^= x; se puede traducir a ensamblador como:**

- > a) xorl %eax,%ebx
- b) xorl %ebx,%eax
- c) andl %ebx,%eax
- d) testl %eax,%ebx

**60. Después de ejecutar una instrucción de suma sobre dos números con signo de los que sabemos que no provocará overflow (los dos números son pequeños en valor absoluto), queremos comprobar si el resultado de la suma es menor que 0. ¿Qué flag necesita comprobar la instrucción de salto condicional equivalente a... ?**

**if (resultado<0) then goto label**

- a) OF
- b) CF
- > c) SF
- d) ZF



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**61. Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.**

```
struct RECORD {
 long value2;
 int ref_count;
 char tag[4];
};

struct NODE {
 double value;
 struct RECORD record;
 char string[8];
};
```

También se declara una variable global "my\_node" como sigue:

```
struct NODE my_node;
```

Si la dirección de my\_node es 0x601040, ¿cuál es el valor de &my\_node.record.tag[1]?

- a) 0x601050
- b) 0x601054
- > c) 0x601055
- d) Ninguna de las anteriores

**62. La instrucción movl %esp,%ebp**

- a) Intercambia los contenidos de los registros ESP y EBP.
- b) Introduce en la pila el contenido del registro EBP.
- > c) Copia el contenido del registro ESP en el registro EBP.
- d) Mueve el contenido del registro ESP al registro EBP, poniendo a 0 el registro ESP.

**63. ¿Cuál de las siguientes listas está correctamente ordenada temporalmente?**

- > a) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
- b) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
- c) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
- d) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX

**64. En IA32 el puntero de pila es:**

- a) una dirección de memoria de 32 bits almacenada en el contador de programa
- b) un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa
- > c) un registro de 32 bits en el microprocesador
- d) un registro de 16 bits en el microprocesador

**65. En un sistema x86-64, si %rsp tiene el valor 0x7fffff0000 inmediatamente antes de ejecutar una instrucción retq, ¿cuál es su valor inmediatamente después?**

- a) 0x7fffff0000
- +0 no: no libera nada
- b) 0x7fffffefff8
- 8 no: pila no libera hacia abajo
- c) 0x7fffff0004
- +4 no: retq retira %rip de 64bits
- > d) 0x7fffff0008
- +8 ok

**66. ¿Cuál de los siguientes fragmentos de código deja en %eax un resultado distinto a los otros tres fragmentos?**

- a) not %eax

add \$1, %eax

//compl.2, definición: neg = not + 1

- b) xor %edx, %edx

sub %eax, %edx

mov %edx, %eax

//0-eax = -eax = compl.2

- > c) mov \$-1, %edx

sub %eax, %edx

mov %edx, %eax

- d) neg %eax

//compl.2

**67. Si AX = 0xFA50 y ejecutamos AND \$0xFF, %AX**

- a) El registro AL se pone a 0
  - b) El registro AL se pone a FF
  - c) El registro AH se pone a FF
- > d) El registro AH se pone a 0

**68. La diferencia entre el flag de acarreo y de overflow es que...**

- > a) ambos se recalculan tras cada operación aritmético-lógica con ints, correspondiendo al programador consultar uno u otro según piense que sus datos son con o sin signo
- b) el de acarreo indica que el resultado es demasiado grande (para p.f.) o positivo (si se trata de ints) para poder almacenarse, el de overflow indica que es demasiado pequeño (p.f.) o negativo (ints)
- c) el flag de acarreo indica que ha habido acarreo en una operación con números enteros (ints), el de overflow indica que ha habido desbordamiento en una operación con números en punto flotante (p.f.)
- d) uno se activa cuando se opera con números con signo y otro cuando son sin signo

**69. Siendo EDX=0xf000 y ECX=0x0100, ¿cuál de las siguientes instrucciones tiene como dirección efectiva 0xf400?**

- > a) xorl (%edx, %ecx, 4), %eax
- b) leal 0x80(%edx, 2), %eax
- c) movl 0x8(%edx), %eax
- d) addl (%edx, %ecx), %eax

**70. Es responsabilidad del procedimiento llamado salvaguardar los registros:**

- a) %eax, %ebx, %ecx, %edx
  - b) %esi, %edi
  - c) %eax, %edx, %ecx
- > d) %ebx, %esi, %edi

**71. ¿Cuál afirmación es FALSA al comparar las arquitecturas x86 y x86-64?**

- a) El tamaño de un double es el mismo
- > b) El tamaño de un puntero es el mismo
- c) El tamaño de un entero (int) es el mismo
- d) El tamaño de las posiciones de memoria es el mismo

**72. Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?**

- a) El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
- b) El desplazamiento D también puede ser el nombre de una variable (que se traduce por su dirección, de 4bytes)
- > c) Los registros base e índice (Rb y Ri) pueden ser cualesquiera de los 8 registros enteros (EAX...ESP)
- d) El factor de escala S puede ser 1, 2, 4, 8

**73. Sean un int\*a y un int n. Si el valor de %ecx es a y el valor de %edx es n, ¿cuál de los siguientes fragmentos de ensamblador se corresponde mejor con la sentencia C return a[n]?**

-> a) mov (%ecx,%edx,4),%eax

ret

b) mov (%ecx,%edx,1),%eax

ret

c) leal (%ecx,%edx,4),%eax

ret

d) ret (%ecx,%edx,4)

**74. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):**

a) movw %dx, (%eax)

-> b) pushb \$0xFF

Tema 2.1, tr.28, Hallaron, p.205-208, Probl. 3.2.5

[CS:APPProb3.2]: Para cada una de las siguientes líneas de lenguaje ensamblador, determinar el sufijo

de instrucción apropiado basándose en los operandos:

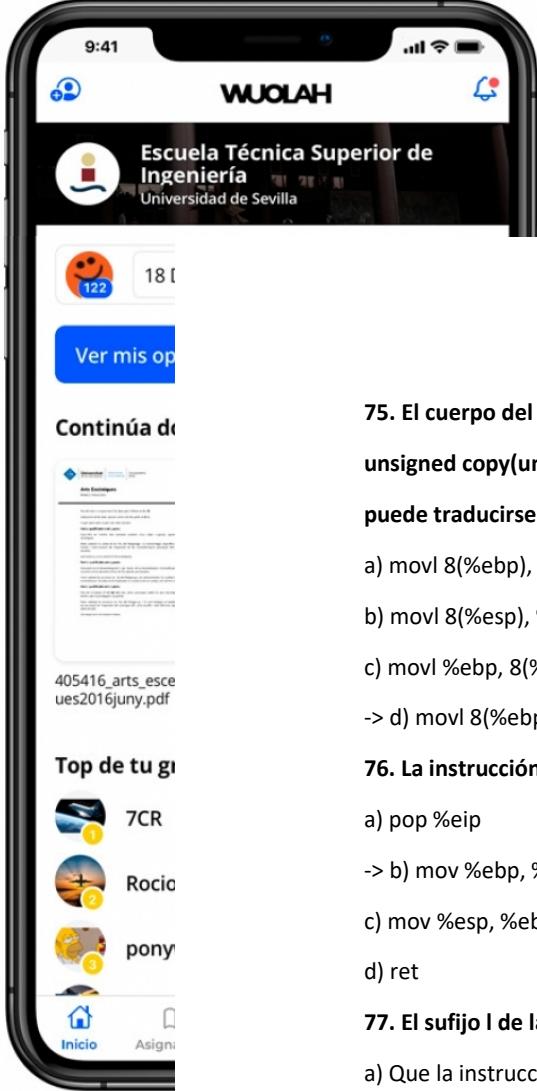
...

5) push \$0xFF

...

c) movswl (%eax), %edx

d) movzbl %dl, %eax



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## 75. El cuerpo del siguiente código C:

`unsigned copy(unsigned u) {return u;}`

puede traducirse a ensamblador como:

- a) `movl 8(%ebp), (%eax)`
- b) `movl 8(%esp), %ebp`
- c) `movl %ebp, 8(%eax)`
- > d) `movl 8(%ebp), %eax`

## 76. La instrucción leave hace exactamente lo mismo que

- a) `pop %eip`
- > b) `mov %ebp, %esp; pop %ebp`
- c) `mov %esp, %ebp; pop %esp`
- d) `ret`

## 77. El sufijo l de la instrucción movl significa:

- a) Que la instrucción usa ordenación de bytes little-endian en lugar de big-endian.
- b) Que la instrucción afecta a la parte de 16 bits más a la izquierda de los operandos (left word).
- > c) Que la instrucción trabaja con operandos de 32 bits (long word).
- d) Que la instrucción afecta a los 16 bits menos significativos de los operandos (low word).

## 78. Comparando las convenciones de llamada de gcc Linux IA32 con x86-64 respecto a registros

- a) En IA32 %esi es salva-invocado, y en x86-64 %rsi es salva-invocado también  
%rsi es 2º argumento - salva-invocante (amarillo en tr.2.4.5)
- b) En IA32 %ebp es especial (marco de pila), y en x86-64 %rbp también  
%rbp es salva-invocado (verde en tr.2.4.5)
- > c) En IA32 %ecx es salva-invocante, y en x86-64 %rcx es salva-invocante también
- d) En IA32 %ebx es salva-invocante, pero en x86-64 %rbx es salva-invocado  
%ebx es salva-invocado (verde en tr.2.3.36)

## 79. ¿Cuál de los siguientes lenguajes no permite el paso de parámetros por referencia?

- a) C++
- b) Pascal
- c) FORTRAN
- > d) C

**80. Dada la siguiente declaración en lenguaje C, una estructura de este tipo podría ocupar, bien sea en un sistema Linux IA32 o bien en uno x86-64, un total de...**

```
struct a{
 int i;
 double d;
 char c;
 short s; };
```

a) 18 B

no, sería 16 B en IA32, sólo 1 B relleno entre c y s

b) 20 B

c) 22 B

-> d) 24 B

**sí, 4 B relleno entre i y d, 1B entre c y s, 4B relleno tras s**

**81. ¿Qué salida produce el siguiente código? Asumir representación de datos de arquitectura IA32.**

```
unsigned int x = 0xDEADBEEF;
unsigned short y = 0xFFFF;
signed int z = -1;

if (x > (signed short) y)
 printf("Hello");

if (x > z)
 printf("World");
```

**(Aviso: en comparaciones con un dato unsigned se pasa el otro dato a unsigned)**

a) Imprime "Hello"

b) Imprime "HelloWorld"

-> c) No imprime nada

En el problema 3.4 sólo se explica que una extensión de tamaño se hace según el fuente sea signed (extensión sgn) o unsigned (ext. con ceros), pero no se explica la sección 2.2.6 (y 2.2.5) en donde se aclara que las comparaciones con unsigned se hacen unsigned. Evitar esta pregunta en el futuro.

d) Imprime "World"

**82. ¿Cuál de los siguientes microprocesadores no es de 64 bits?**

- a) Core 2
- > b) Pentium III
- c) Core i7
- d) Itanium

**83. Si el registro %eax contiene el siguiente valor binario:**

**11111111 10101010 01010101 11110000**

**¿Cuál será el valor de %eax tras ejecutar la instrucción xorb %al, %al?**

- a) 00000000 10101010 01010101 11110000
- > b) 11111111 10101010 01010101 00000000
- c) 11111111 10101010 01010101 11110000
- d) 00000000 00000000 00000000 00000000

**84. Respecto a los registros enteros en arquitectura IA32**

- > a) Hay 8 de cada tamaño (32, 16, 8 bits), aunque no todos los registros tienen versión en 8 y 16 bits

Tema1 tr.26

- b) Son de 32bits en las aplicaciones de 32bit, y de 64bits en las aplicaciones de 64bit

- c) No hay distintos tamaños, son sólo registros de 32 bits, como corresponde a dicha arquitectura

- d) Hay 8, y en cada uno puede accederse a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX) ó a los 8 LSBs (p.ej. AL)

**85. Para traducir una construcción if-then-else de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente**

- > a) un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional

- b) dos saltos condicionales y dos saltos incondicionales

- c) dos saltos condicionales (uno para la parte if y otro para la parte else)

- d) un salto condicional, según la condición expresada en el código C

**86. ¿Cuál afirmación es FALSA en arquitecturas x86-64?**

- a) El tamaño de un puntero es 64 bits

- b) El tamaño de los registros es 64 bits

- > c) El tamaño de las posiciones de memoria es 64 bits

- d) El tamaño de un double es 64 bits

**87. Si A y B son dos enteros almacenados respectivamente en %eax y %ebx, ¿cuál de las siguientes implementaciones de if (!A && !B) {...then part...} es incorrecta?**

-> a) test %ebx, %eax

jne not\_true

...then part...

not\_true:

...

test hace AND, no OR

equivale a if (A&&B) {... then\_part ...}

b) or %ebx, %eax

jne not\_true

...then part...

not\_true:

...

de Morgan

equivale a if ( !(A|B) ) {... then\_part ...}

c) test %eax, %eax

jne not\_true

test %ebx, %ebx

jne not\_true

...then part...

not\_true:

...

short-circuit

gcc usa test %eax,%eax para comprobar cmp \$0,%eax

d) cmp \$0, %eax

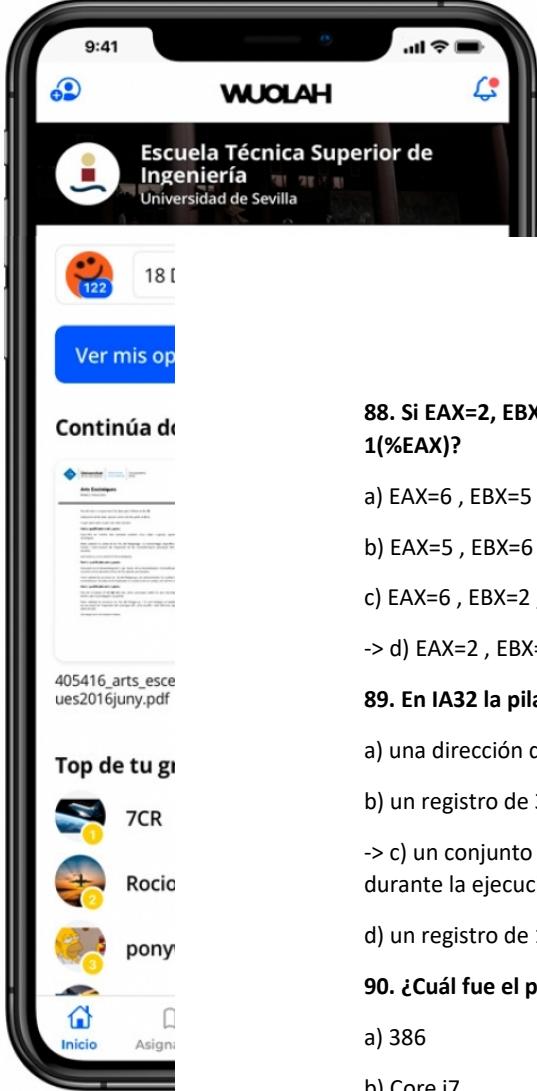
jne not\_true

cmp \$0, %ebx

jne not\_true

...then part...

not\_true:



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**88. Si EAX=2, EBX=5 y M[3]=3, ¿qué valores quedan tras ejecutarse la instrucción XOR %BL, 1(%EAX)?**

- a) EAX=6 , EBX=5 , M[3]=2
- b) EAX=5 , EBX=6 , M[3]=2
- c) EAX=6 , EBX=2 , M[3]=5
- > d) EAX=2 , EBX=5 , M[3]=6

**89. En IA32 la pila es:**

- a) una dirección de memoria de 32 bits almacenada en el contador de programa
- b) un registro de 32 bits en el microprocesador
- > c) un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa
- d) un registro de 16 bits en el microprocesador

**90. ¿Cuál fue el primer procesador de Intel de 64-bits en la familia x86(-64)?**

- a) 386
- b) Core i7
- c) 8086
- > d) Pentium 4F

**91. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?**

- a) EBX
- b) EDI
- > c) EAX
- d) ESI

**92. Para traducir una construcción do-while de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente**

- > a) un salto condicional hacia atrás, según la misma condición que en lenguaje C
- Tema 2.2, tr.40
- b) un salto condicional hacia atrás, según la condición opuesta a la de lenguaje C
- c) un salto condicional hacia adelante, según la condición opuesta a la de lenguaje C
- d) un salto condicional hacia adelante, según la misma condición que en lenguaje C

**93. ¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?**

a) CALL dir

b) JMP dir

c) RET

-> d) JNE dir

**94. Estudiando el listado de una función C presuntamente compilada con gcc en modo 64bit (x86-64), nos dicen que la instrucción "movl (%rdi), %eax" carga en el registro EAX el valor adonde apunta el primer argumento.**

a) Está mal, porque el primer argumento de una función C no se pasa en RDI

b) Está mal, porque EAX no se puede usar en modo 64bit, debería ser RAX

-> c) Está bien, y pone a cero los 32 bits más significativos de RAX

d) Está mal, porque EAX no se carga con ningún valor

**95. En el direccionamiento inmediato el operando reside en:**

-> a) en la instrucción tras el código de operación

b) en la pila

c) en un registro del procesador

d) en memoria, en la dirección indicada

**96. Sobre el direccionamiento relativo a contador de programa:**

a) Favorece la implementación de código reubicable.

b) Su uso en los saltos y llamadas a subrutinas reduce el tamaño de la instrucción.

c) Es adecuado para alcanzar instrucciones próximas a la que se está ejecutando.

-> d) Todas las respuestas son ciertas.

**97. Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA32 como en x86-64. Sólo una de ellas es FALSA.**

**¿Cuál?**

-> a) El tamaño de un long es 4 bytes

b) El tamaño de un int es 4 bytes

c) El tamaño de un unsigned es 4 bytes

d) El tamaño de un short es 2 bytes

**98. Si EAX=2, EBX=5 y M[3]=3, ¿qué valores quedan tras ejecutarse la instrucción XOR %BL, 1(%EAX)?**

- a) EAX=6 , EBX=5 , M[3]=2
- b) EAX=5 , EBX=6 , M[3]=2
- > c) EAX=2 , EBX=5 , M[3]=6
- d) EAX=6 , EBX=2 , M[3]=5

**99. ¿Cuál de las siguientes afirmaciones sobre la instrucción leave es cierta?**

- a) Equivale a mov %esp,%ebp seguida de pop %ebp //mov al revés

b) Se ejecuta justo después de retornar de un procedimiento

lo típico después de retornar (de una función C) es pop/add \$n,%esp y/o mov %eax,...

- c) Equivale a pop %ebp seguida de mov %ebp,%esp

en realidad es mov %ebp,%esp; pop %ebp

-> d) No es obligatorio usarla. En su lugar puede realizarse una secuencia explícita de operaciones mov y pop

**100. En el fragmento de código**

**804854e:e8 3d 06 00 00 call 8048b90 <main>**

**8048553:50                    pushl %eax**

**la instrucción call suma al contador de programa la cantidad:**

- a) 0x0804854e

-> b) 0x0000063d

- c) 0x08048553

- d) 0x50

**101. Alguna de las siguientes afirmaciones sobre sistemas Linux x86-64 no es cierta**

-> a) Todos los argumentos de función se pasan a través de la pila

b) %rax se usa para devolver los valores de retorno de funciones

c) %eax y %ebx pueden usarse como en un sistema IA32

d) %rbp se puede usar como cualquier otro registro (no hay puntero base)

**102. ¿Cuál de los siguientes no es un modo de direccionamiento IA32?**

-> a) Cache

b) Registro

c) Memoria

d) Inmediato

**103. Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.**

```
struct RECORD {
 long value2;
 int ref_count;
 char tag[4];
};

struct NODE {
 double value;
 struct RECORD record;
 char string[8];
};
```

También se declara una variable global "my\_node" como sigue:

```
struct NODE my_node;
```

¿Cuál es el tamaño de my\_node en bytes?

a) 28

-> b) 32

struct record 8+4+4=16B 8x, struct node 8+16+8=32B 8x

c) 40

d) Ninguno de los anteriores

**104. El registro SP / ESP / RSP...**

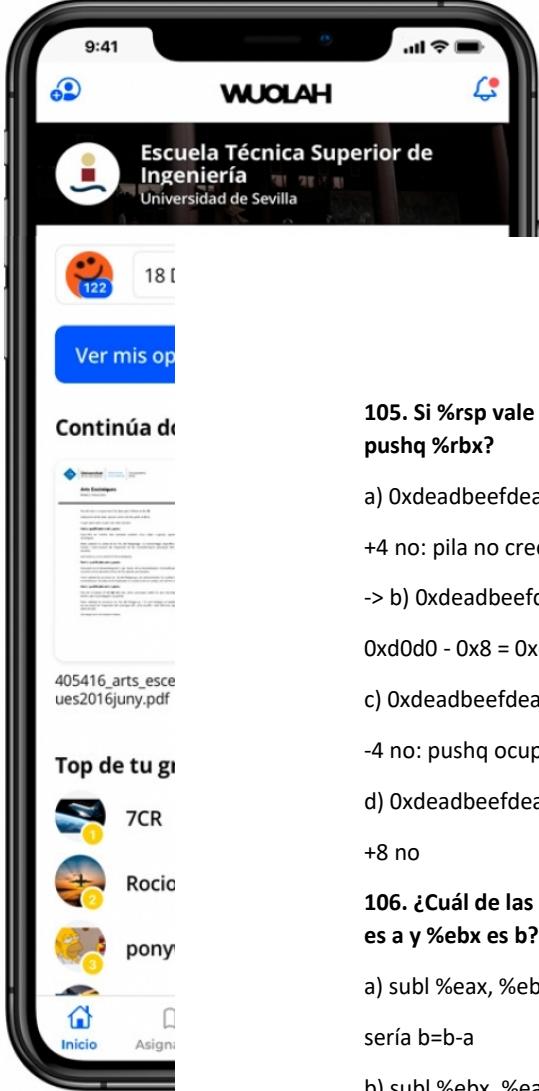
a) es un registro transparente al usuario y contiene la dirección de memoria a la que se está accediendo

sería MAR o el equivalente en IA32

b) es un registro transparente al usuario y contiene la instrucción que se está ejecutando  
ni siquiera IP apunta a la instrucción en ejecución, sino a la siguiente

-> c) es un registro de propósito específico y contiene la dirección de la cima de la pila  
específico en el sentido de que no se puede usar como índice y se usa implícitamente (sin nombrarlo) en push, pop, call, ret...

d) es un registro de propósito específico y contiene la dirección de la siguiente instrucción a ejecutar



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**105.** Si %rsp vale Oxdeadbeefdeadd0d0, ¿cuál será su nuevo valor después de que se ejecute pushq %rbx?

- a) Oxdeadbeefdeadd0d4
- +4 no: pila no crece hacia arriba
- > b) Oxdeadbeefdeadd0c8
- $0xd0d0 - 0x8 = 0xd0c8$
- c) Oxdeadbeefdeadd0cc
- 4 no: pushq ocupa quad word
- d) Oxdeadbeefdeadd0d8
- +8 no

**106.** ¿Cuál de las siguientes secuencias de instrucciones calcula a=b-a, suponiendo que %eax es a y %ebx es b?

- a) subl %eax, %ebx
- sería  $b=b-a$
- b) subl %ebx, %eax
- sería  $a=a-b$
- c) notl %eax
- $addl %ebx, %eax$
- sería  $a=-a-1+b=b-a-1$

-> d) Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una

**107.** En un sistema de 32bits, ¿cuál de las siguientes expresiones C es equivalente a la expresión  $(x[2] + 4)[3]$ ?

(Asumir que x se ha declarado como int \*\*x. Recordar que C usa aritmética de punteros. Notar que muchos de los paréntesis no son necesarios, sólo se han añadido para evitar confusiones por precedencia de operadores)

- > a)  $*((x + 2)) + 7$
- ok,  $(X[2])[7]$
- b)  $*(((*x) + 2) + 7)$
- sería  $X[0][2+7]$
- c)  $(**x + 2) + 7$
- sería  $X[2][0]+7$ , la última suma sería suma entera, no aritmética puntero
- d)  $*((x + 8)) + 28$

**108. ¿Cuál de las siguientes instrucciones x86 se puede usar para sumar dos registros y guardar el resultado sin sobrescribir ninguno de los registros originales?**

a) mov

-> b) lea

c) add

d) Ninguna de ellas

**109. La instrucción test es...**

a) Lo mismo que and

b) Lo mismo que sub, pero no guarda el resultado, sólo ajusta los flags

c) Lo mismo que sub

-> d) Lo mismo que and, pero no guarda el resultado, sólo ajusta los flags

**110. La primera letra (l) de la instrucción lea:**

a) indica que la instrucción afecta a los 16 bits menos significativos del operando destino (low word)

b) indica que la instrucción usa ordenación de bytes little-endian

c) indica que la instrucción trabaja con un operando destino de 32 bits (long word)

-> d) forma parte del nemotécnico de la instrucción

**111. ¿Cuál de las siguientes instrucciones es incorrecta en ensamblador GNU/as IA32?**

a) mov (%esp),%ebp

-> b) pop %eip

No existe pop %eip, usar ret para conseguir ese objetivo.

El problema 3.30 avisa de que call/pop es la única forma de copiar %eip, sugiriendo que no se puede con mov/lea ni con push/pop.

c) lea 0x10(%esp),%ebp

d) pop %ebp

**112. Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:**

a) simulador

b) ensamblador

-> c) compilador

d) intérprete

**113. En la secuencia de programa siguiente:**

804854e:e8 3d 06 00 00 call 8048b90 <main>

8048553:50                    pushl %eax

**¿cuál es el valor que introduce en la pila la instrucción call?**

- a) 0x8048b90
- b) 0x804854e
- c) 0x804854f
- > d) 0x8048553

**114. La instrucción necesaria para cargar 0x07 en %eax es:**

- a) movl \$0x07,%ah
- > b) movl \$0x07,%eax
- c) movb \$0x07,%al
- d) movb \$0x07,%eax

**115. La diferencia entre las instrucciones test y cmp consiste en que**

- a) test realiza una operación and lógico, mientras que cmp realiza una resta
- b) test modifica sólo los flags lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
- > c) ambas respuestas son correctas
- d) ambas respuestas son incorrectas

**116. La dirección efectiva del primer parámetro de llamada a una función suele calcularse desde el código de la función como:**

- a) EBP-8
- b) EBP-4
- > c) EBP+8
- d) EBP+4

**117. ¿Cuál es el popcount (peso Hamming, nº de bits activados) del número 29?**

- a) 2
- b) 3
- > c) 4

$29 = 0x1D = 16+8+4+1 = 0b0001\ 1101 \rightarrow \text{popcount } 4$

- d) 5

**118. Un procesador cuya instrucción CALL guardara la dirección de retorno en un registro RL (llamado "de enlace"):**

- a) Permitiría llamadas anidadas y recursivas.
- b) Permitiría llamadas recursivas, pero no anidadas.
- c) Permitiría llamadas anidadas, pero no recursivas.
- > d) No permitiría llamadas anidadas ni recursivas.

**119. Si %edx contiene 0xf000 y %ecx contiene 0x0100, el direccionamiento 0x80(%ecx,%edx,2) se refiere a la posición**

- a) 0xf182
- b) 0xf280
- > c) 0x1e180
- d) Ninguna de las respuestas anteriores es correcta

**120. Si la variable val está almacenada en ebx y la variable x está almacenada en eax, la sentencia val ^= x; se puede traducir a ensamblador como:**

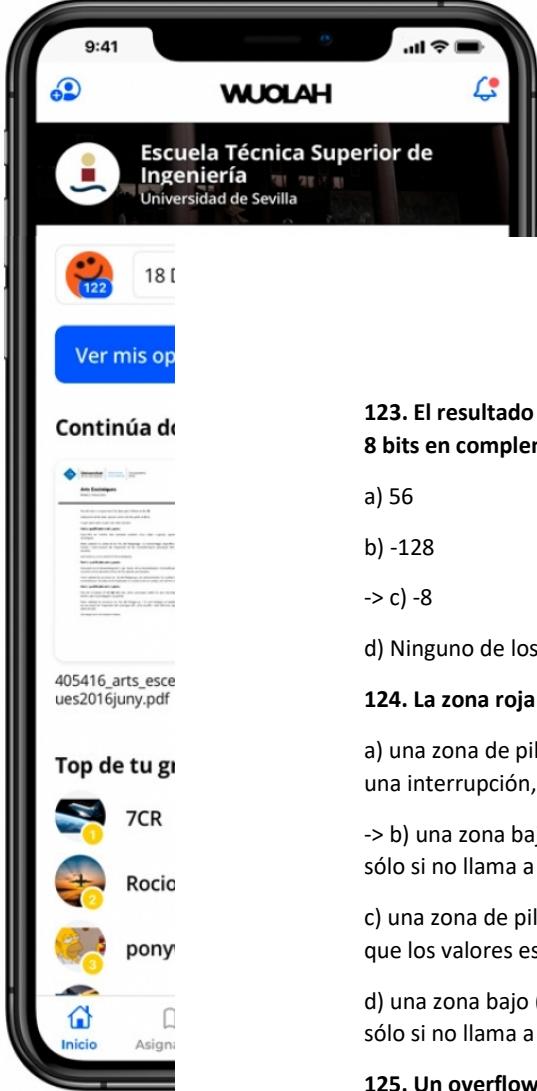
- a) testl %eax,%ebx
- > b) xorl %eax,%ebx
- c) andl %ebx,%eax
- d) xorl %ebx,%eax

**121. Un archivo .o que contiene código objeto:**

- a) Contiene las direcciones definitivas de las variables globales.
- b) Incluye el código de las funciones de biblioteca a las que llame.
- > c) Contiene instrucciones máquina binarias.
- d) Puede ejecutarse directamente.

**122. Alguna de estas opciones contiene algún elemento que no corresponde a los contenidos de un marco de pila GCC/Linux IA32.**

- a) Argumentos de llamada a la función y Dirección de retorno
- > b) Variables globales y Valor de retorno de la función
- c) Variables locales y Registros salva-invocado
- d) Antiguo marco de pila y Registros salva-invocante



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**123. El resultado de desplazar aritméticamente dos posiciones hacia la derecha el número de 8 bits en complemento a dos -32 es:**

- a) 56
- b) -128
- > c) -8
- d) Ninguno de los resultados anteriores es correcto

**124. La zona roja en x86-64 Linux es...**

- a) una zona de pila en donde no deben escribir las funciones invocadas (porque si se produjera una interrupción, el manejador de interrupción sobreescribiría los valores escritos en pila)
- > b) una zona bajo (RSP) (adonde apunta RSP) que una función puede usar sin reservarla, pero sólo si no llama a ninguna otra función mientras la usa
- c) una zona de pila en donde pueden escribir las funciones invocadas, pero teniendo en cuenta que los valores escritos pueden verse alterados si se produce una interrupción
- d) una zona bajo (RBP) (adonde apunta RBP) que una función puede usar sin reservarla, pero sólo si no llama a ninguna otra función mientras la usa

**125. Un overflow nunca puede ocurrir cuando:**

- a) se suman dos números positivos  
 $0x7fffffff + 0x1$
- > b) se suma un número positivo a un número negativo  
resultado queda entre ambos => se puede representar
- c) se suman dos números negativos  
 $0x80000000 + 0xffffffff$
- d) se resta un número positivo de un número negativo  
 $0x80000000 - 0x00000001$

**126. ¿En qué registro se pasa el primer argumento a una función según el estándar \_cdecl en una arquitectura IA32?**

- a) edi
- b) esi
- c) eax
- > d) Las anteriores respuestas son erróneas  
se pasa en registros en x86-64

**127. GCC/Linux IA32 resuelve el ajuste de marco de pila mediante las instrucciones:**

- > a) pushl %ebp; movl %esp, %ebp
- b) movl %ebp, %esp; popl %ebp
- c) pushl %esp; movl %ebp, %esp
- d) movl %esp, %ebp; popl %esp

**128. Si ECX vale 0, la instrucción adc \$-1,%ecx**

- a) Pone CF=1
- b) Cambia CF
- c) Pone CF=0
- > d) No cambia CF

**129. ¿Cuál de las siguientes parejas de mnemotécnicos de ensamblador IA32 corresponden a la misma instrucción máquina?**

- a) CMP (comparar), SUB (restar)
- > b) JZ (saltar si cero), JE (saltar si igual)
- c) JC (saltar si acarreo), JL (saltar si menor, para números con signo)
- d) SAR (desplazamiento aritmético a la derecha) / SHR (desplazamiento lógico a la derecha)

**130. ¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una función long int al terminar ésta?**

- a) La instrucción RET lo almacena en un registro especial de retorno.
- b) Por convención se guarda en %eax.
- c) Se almacena en pila justo encima de los argumentos de la función.
- > d) Ninguna de esas formas es la correcta

**131. Considere una función C declarada así:**

```
void fun4arg(int a, int b, int c, int d);
```

**Suponiendo que fun4arg se ha compilado para una máquina x86 IA32 con enteros de 4 bytes, ¿cuál sería la dirección del argumento b relativa a %ebp, en el marco de pila de fun4arg?**

- a) %ebp + 8
- b) %ebp + 16
- > c) %ebp + 12
- d) %ebp + 20

**132. Para desplazar %eax a la derecha un número variable de posiciones <= 32, indicado en %ebx, se puede hacer**

a) sar %ebx, %eax

no existe

-> b) mov %ebx, %ecx

sar %cl, %eax

ver problema Hallaron 3.8 y sección 3.5.3

c) sar %bl, %eax

no existe

d) No se puede, sar sólo admite un número fijo de posiciones (debe ser un valor inmediato)

**133. Respecto a la convención de llamada usada en Linux/gcc:**

a) Una subrutina que modifique algún registro debe restaurar su valor anterior antes de retornar

b) Todos los registros pueden ser modificados libremente por todas las subrutinas

-> c) Hay registros que pueden ser modificados libremente por las subrutinas, y otros que, si se modifican, se deben restaurar posteriormente. Y también hay registros especiales

d) Hay registros modificables, otros que deben ser restaurados, y las subrutinas anidadas deben respetar los registros modificables que están en uso por otras subrutinas

**134. Si la dirección del primer elemento de un vector de enteros z está almacenada en el registro %edx y la variable entera i está almacenada en el registro %eax, la instrucción máquina que realiza la operación z[i]++ es:**

a) addl \$4, (%eax,%edx)

no, z[i]++ suma 1, no 4

-> b) addl \$1, (%edx,%eax,4)

c) addl \$1, (%edx,%eax)

no, falta multiplicar ix4 para indexar en array enteros

d) addl \$1, (%eax,%edx,4)

no, (dirección de z)x4 no

**135. ¿Cuál de las siguientes afirmaciones es incorrecta?**

a) En lenguajes de alto nivel no se tiene acceso a detalles del procesador como las tablas de páginas, el paso de modo usuario a modo supervisor, el bit de paridad, etc.

b) En lenguaje ensamblador cada instrucción se corresponde con una instrucción máquina.

-> c) En lenguaje ensamblador las instrucciones se escriben en binario.

d) El lenguaje de alto nivel es más portable que el lenguaje máquina.

**136. Para comprobar si el entero almacenado en EAX es cero (y posiblemente saltar a continuación usando JZ/JNZ), gcc genera el código**

a) test %eax

test tiene 2 argumentos

b) cmp %eax, \$0

en todo caso sería cmp \$0,%eax

no le gusta a gcc eso

-> c) test %eax, %eax

d) cmp %eax

cmp tiene 2 argumentos

**137. Cuando se ejecuta la instrucción ret al final de una subrutina:**

-> a) la dirección almacenada en la cima de la pila se transfiere al contador de programa

b) la dirección almacenada en la cima de la pila se transfiere al puntero de pila

c) la dirección de comienzo de la pila se transfiere al puntero de pila

d) la dirección de memoria de la instrucción ret se transfiere al contador de programa

**138. Considerar la declaración C**

`int array[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};`

**Suponer que el compilador tiene la dirección de array en el registro %ecx. ¿Cómo se movería el valor array[3] al registro %eax? Asumir que %ebx es 3.**

-> a) movl (%ecx,%ebx,4),%eax

b) movl 8(%ecx,%ebx,2),%eax

c) leal 12(%ecx),%eax

d) leal (%ecx,%ebx,4),%eax

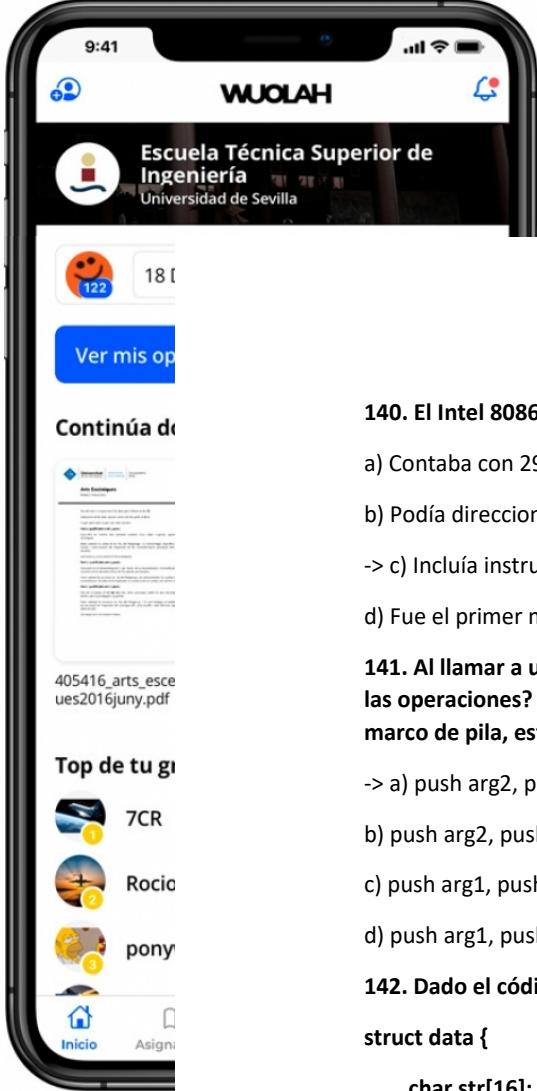
**139. La instrucción xor \$3, %eax tiene como resultado:**

a) Poner a 0 los últimos 3 bits del registro EAX

-> b) Cambiar 0<->1 (complemento a 1 de) los últimos 2 bits del registro EAX

c) Poner a 1 el último bit del registro EAX

d) Ninguno de los anteriores resultados



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

## 140. El Intel 8086:

- a) Contaba con 29 millones de transistores.
- b) Podía direccionar 1 GB.
- > c) Incluía instrucciones de multiplicación.
- d) Fue el primer microprocesador.

## 141. Al llamar a una función de 2 argumentos foo(arg1, arg2), ¿cuál es el orden correcto de las operaciones? (suponiendo convención de llamada x86 cdecl, y que foo requiere ajustar marco de pila, esto es, salvar %ebp)

- > a) push arg2, push arg1, call foo, push %ebp
- b) push arg2, push arg1, push %ebp, call foo
- c) push arg1, push arg2, call foo, push %ebp
- d) push arg1, push arg2, push %ebp, call foo

## 142. Dado el código C siguiente:

```
struct data {
 char str[16];
};

char *f(struct data *ptr) {
 return &(ptr->str[2]);
}
```

la función se traducirá a ensamblador de x86-64 como:

- a) movq (%rdi,2), %rax; ret
- b) leaq (%rdi,2), %rax  
ret
- > c) leaq 2(%rdi), %rax  
ret
- d) movq 2(%rdi), %rax  
ret

## 143. En IA32, el registro contador de programa se denomina:

- > a) EIP
- b) PCR
- c) RIP
- d) PC

**144. Respecto a tamaños de tipos integrales en x86 y x86-64, la excepción es que**

- a) int pasa de 4 B (x86) a 8 B (x86-64)
- > b) long int pasa de 4 B a 8 B
- c) long long int pasa de 4 B a 8 B
- d) ninguna de las anteriores

**145. Si tenemos un número n, de 64 bits, almacenado en la pareja de registros EDX:EAX (EDX contiene los 32 bits más significativos y EAX los 32 bits menos significativos) y queremos realizar la división n / (2^32) entonces:**

- a) Podemos usar las instrucciones siguientes, pero sólo en el caso de que n sea un número sin signo:

```
mov $0x100000000,%ecx
```

```
div %ecx
```

- b) Podemos quedarnos con EDX, pero sólo en el caso de que n sea un número sin signo.

- c) Podemos usar las instrucciones siguientes tanto si n es un número con signo como sin signo:

```
mov $0x100000000,%ecx
```

```
div %ecx
```

- > d) Podemos quedarnos con EDX tanto si n es un número con signo como sin signo.

Cocientes positivos o sin signo redondeados hacia cero, cocientes negativos redondeados hacia más negativo: es decir, redondeado siempre hacia inferior.

No es la división entera estándar de C (que redondea hacia cero) pero el enunciado tampoco lo exige.

**146. La extensión de signo a m bits de un número original N de n bits, con m > n, consiste en:**

- a) Realizar la operación  $2^m - N - 1$

- b) Incrementar la cantidad de bits a m rellenando con unos por la izquierda.

- c) Realizar la operación  $2^m - N$

- > d) Incrementar la cantidad de bits a m preservando el signo y el valor del número.

**147. En un sistema Linux x86-64, ¿cuál de las siguientes variables ocupa más bytes en memoria?**

- a) char a[7]

- > b) int \*c

- c) short b[3]

- d) float d

**148. ¿Cuál de las siguientes secuencias de instrucciones multiplica el (contenido del) registro EAX por 18?**

a) sarl \$1, %eax

imul \$9, %eax

sería shl

b) imull \$0x18, %eax

sería \$18 sin 0x hex

c) shll \$18, %eax

equivale a  $x2^{18}$

-> d) leal (%eax,%eax,8), %eax

leal (%eax,%eax), %eax

**149. En los casos concretos indicados para las siguientes instrucciones IA32, ¿cuál no funciona como instrucción de transferencia?**

a) pushl %eax

-> b) cmpl 0x08048040, %eax

c) movl \$0x15, %eax

d) leal variable, %eax

**150. En un sistema IA32 Linux, ¿cuál es el tamaño de un long?**

a) 8 bytes

-> b) 4 bytes

c) 6 bytes

d) 2 bytes

**151. ¿Cuál de las siguientes secuencias de instrucciones multiplica %eax por 10?**

a) leal(%eax,%eax,4), %eax

sall \$2, %eax

sería  $x5 \times 4 = x20$

b) imull \$0x10, %eax //sería  $x16$

c) addl %eax, %eax

shll \$5, %eax

sería  $x2 \times 32 = x64$

-> d) Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una

**152. Estudiando el listado de una función C presuntamente compilada con gcc en modo 64bit (x86-64), nos dicen que la primera instrucción, movl %eax, (%rdi), carga en EAX el valor adonde apunta el primer argumento.**

- a) Está mal, porque EAX no se puede usar en modo 64bit, debería ser RAX
- b) Está bien, y pone a cero los 32 bits más significativos de RAX
- > c) Está mal, porque EAX no se carga con ningún valor
- d) Está mal, porque el primer argumento de una función C no se pasa en RDI

**153. La instrucción cmovb %edx, %eax**

- a) copia en %eax el contenido de %edx si %eax es menor que %edx  
haría falta cmp %edx, %eax previo para ajustar CF
- b) copia en %eax el byte de memoria apuntado por la dirección contenida en %edx  
inventado, sería (%edx) y además %eax es de tamaño l (long)
- > c) copia en %eax el contenido de %edx si el indicador de acarreo es 1  
"below" equivale a CF
- d) copia el byte bajo de %edx en el byte bajo de %eax

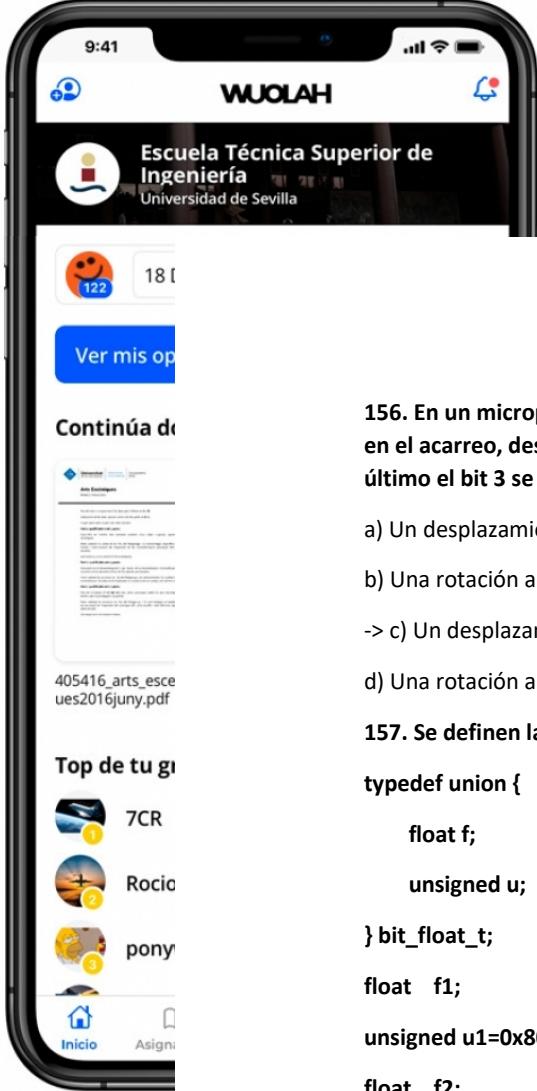
**154. Una función C llamada get\_el() genera el siguiente código ensamblador. Se puede adivinar que:**

```
movl 8(%ebp), %eax
leal (%eax,%eax,4), %eax
addl 12(%ebp), %eax
movl var(%eax,4), %eax
```

- > a) var es un array bidimensional de enteros, con cinco columnas
- b) var es un array bidimensional de enteros, no se pueden adivinar dimensiones
- c) var es un array multi-nivel (punteros a enteros) de cuatro filas
- d) var es un array multi-nivel pero no se pueden adivinar las dimensiones

**155. ¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?**

- a) En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
- b) Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero
- c) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable
- > d) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

**156.** En un microprocesador de 4 bits, una operación en la que el bit 0 de un registro se copia en el acarreo, después el bit 1 se copia en el bit 0, después el bit 2 se copia en el bit 1, y por último el bit 3 se copia en el bit 2, es:

- a) Un desplazamiento lógico a la derecha.
- b) Una rotación a la derecha a través de acarreo.
- > c) Un desplazamiento aritmético a la derecha.
- d) Una rotación a la derecha.

**157.** Se definen la unión, variables, y función C, siguientes:

```
typedef union {
 float f;
 unsigned u;
} bit_float_t;

float f1;

unsigned u1=0x80000000;

float f2;

float bit2float(unsigned u) {
 bit_float_t arg;
 arg.u = u;
 return arg.f;
}
```

**¿Cuál afirmación es verdadera?**

- a) Si asignamos f1=bit2float(u1); f2=float(u1); entonces f1==f2, ambos distintos de cero  
float(u1) es error de compilación, y además f1 no es distinto de cero
- b) Si asignamos f1=bit2float(u1); f2=(float)u1; entonces f1== 0.0 pero f2==4294967296.00  
0x8000 0000 son 2G, no 4G, f2 está mal
- c) Si asignamos f1=bit2float(u1); f2=float(u1); entonces f1==1073741824.00 y f2==  
-1073741824.00  
float(u1) es error de compilación, y además +-1G no es 0x8000 0000
- > d) Si asignamos f1=bit2float(u1); f2=(float)u1; entonces f1== -0.0 pero f2==2147483648.00  
correcto, 0x8000 0000 es el bit de signo, y también es 2G

**158. Si el contenido de EBP es 0x13000, ¿a qué dirección se hace referencia con la instrucción INC –0x80(%EBP)?**

- a) 0x13080
- > b) 0x12F80
- c) 0x13000
- d) 0x80

**159. Para poner a 1 el bit 5 del registro %edx sin cambiar el resto de bits podemos usar la instrucción máquina:**

- > a) or \$0x20, %edx
- b) and \$0x5, %edx
- c) and \$32, %edx
- d) or \$0b101, %edx

**160. En la nomenclatura del ensamblador de IA32, una cantidad de 16 bits es designada como:**

- a) half word
- > b) word
- c) double word
- d) quad Word

**161. Si declaramos int val[5]={1,5,2,1,3}; entonces**

- a) val[5] es de tipo int y vale 3

val[4]==3, val[5] apunta fuera del array

- b) val+1 es de tipo int y vale 2

val+1==&val[1] es int\*

- > c) &val[2] es de tipo int\* y vale lo mismo que (void\*)val+8

En Sep15 faltaba (void\*) y entonces sería falsa por aritmética de punteros

- d) val+4 es de tipo int\* y se cumple que \*(val+4)==5

Por aritmética de punteros, \*(val+4)==val[4]==3

**162. ¿Cuál sería el "equivalente x86-64" del "pseudo-código C" `rcx = ((int*)rax)[rcx]`?**

- a) `lea (%rax,4,%rcx),%rcx`
- b) `lea (%rax,%rcx,4),%rcx`
- c) `mov (%rax,4,%rcx),%rcx`
- > d) `mov (%rax,%rcx,4),%rcx`

eso sería `movq`, debería ser `movl` destino `%ecx`

si se indexa en `int[]` se debería obtener `int`, no `long`, si se hubiera dicho `(long*)rax` tendríamos que indexar `8x`, no `4x`, evitar esta pregunta

**163. Una instrucción de "salto si igual" tiene que comprobar el valor de:**

- a) los bits de signo y desbordamiento
- b) el bit de signo
- c) el bit de acarreo
- > d) el bit de cero

**164. La instrucción IA32 test sirve para...**

- > Realizar la operación and lógico bit-a-bit (`a&b`) pero no guardar el resultado, sino simplemente ajustar los flags
- b) Realizar la operación resta (`a-b`) pero no guardar el resultado, sino simplemente ajustar los flags
- c) Testear el código de condición indicado, y poner un byte a 1 si se cumple
- d) Mover el operando fuente al destino, pero sólo si se cumple la condición indicada

**165. Dentro de una función declarada como `void swap(int *xp, int *yp)`, que intercambia los valores de los dos enteros cuyas direcciones de memoria (punteros) son pasadas como parámetros a la función, la instrucción `movl 12(%ebp),%ecx` copia en `%ecx`...**

- a) el valor del entero apuntado por el puntero pasado como primer parámetro
- > b) el valor del puntero pasado como segundo parámetro
- c) el valor del puntero pasado como primer parámetro
- d) el valor del entero apuntado por el puntero pasado como segundo parámetro

**166. ¿Qué combinación de flags aritmético-lógicos corresponde al código de condición b (below)?**

- > a) CF
- b) OF xor SF
- c) CF xor OF
- d) OF

**167. Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.**

```
struct RECORD {
 int value2;
 short ref_count;
 char tag[10];
};

struct NODE {
 long value;
 struct RECORD record;
 char string[8];
};
```

También se declara una variable global "my\_node" como sigue:

```
struct NODE my_node;
```

Si la dirección de my node es 0x600940, ¿cuál es el valor de &my\_node.record.tag[1]?

- a) 0x60094a
- b) 0x60094e
- > c) 0x60094f
- d) Ninguna de las anteriores

**168. Si RCX vale -1, tras ejecutar las instrucciones**

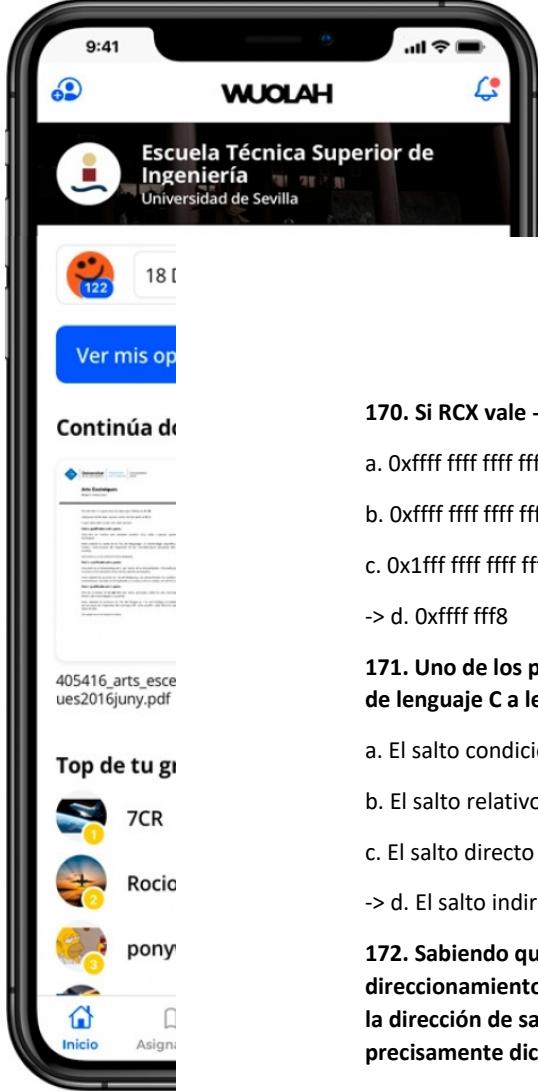
```
rol $1, %cl
rcr $2, %rcx
```

el nuevo valor de RCX y del flag CF es

- a. hay algún fallo de sintaxis o gramática en esas instrucciones
- b. RCX≠-1, CF mantiene su valor
- > c. RCX=-1, CF=1 el primer rol pone CF=1, el segundo recircula pero todos los bits son 1
- d. no se puede marcar ninguna de las opciones anteriores

**169. Para comprobar si el entero almacenado en RAX es cero (y tal vez saltar a continuación usando JZ/JNZ), gcc genera el código**

- a. cmp %eax, \$0 en todo caso sería cmp \$0,%eax - no le gusta a gcc eso
- b. test %eax test tiene 2 argumentos
- c. cmp %eax cmp tiene 2 argumentos
- > d. test %eax, %eax



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**170. Si RCX vale -1, tras ejecutar la instrucción sal \$3, ¿cuál es el nuevo valor de RCX es**

- a. Oxffff ffff ffff ffff
- b. Oxffff ffff ffff fff8
- c. Ox1fff ffff ffff ffff
- > d. Oxffff ffff8

**171. Uno de los puntos clave de la traducción que gcc hace de una construcción switch-case de lenguaje C a lenguaje ensamblador es...**

- a. El salto condicional hacia atrás
- b. El salto relativo a contador de programa
- c. El salto directo
- > d. El salto indirecto

**172. Sabiendo que las instrucciones de salto condicional codifican la dirección de salto con direccionamiento relativo a contador de programa (de 8 o 32 bits con signo), indicar cuál es la dirección de salto de la instrucción je en el siguiente desensamblado, donde se ha tachado precisamente dicha dirección.**

**40042f: 74 f4 je xxxxxx**

**400431: 5d pop %rbp**

- a. 400431 eso sería si el desplazamiento relativo fuera 0
- b. 400525 eso es 0x400431 + 0xf4, se debe extender signo
- > c. 400425 Ej.3.15b p.245/367 del libro, y Tema2.3, tr.15-16
- d. 40043d

**173. Para traducir una asignación condicional (a=b?c:d;) de lenguaje C a lenguaje ensamblador, gcc puede que utilice...**

- a. Un salto incondicional, según la condición expresada en el código C, y otro salto incondicional
- b. Un salto condicional, según la condición opuesta a la del código C, y otro salto condicional incond.,
- > c. Una instrucción de movimiento condicional, pero sólo si el procesador es Pentium Pro/II o superior
- d. Una instrucción de movimiento incondicional, pero sólo si el S.O. es de 64bits

**174. Sabiendo que las instrucciones de salto condicional codifican la dirección de salto con direccionamiento relativo a contador de programa (de 8 o 32 bits con signo), indicar cuál es la dirección de la instrucción pop en el siguiente desensamblado, donde se ha tachado la parte de las direcciones.**

xxxxxx: 77 02 ja 400547

xxxxxx: 5d pop %rbp

a. 400547

b. 400549

-> c. 400545

d. 400543

**175. En la convención de llamada SystemV AMD64 seguida por gcc Linux/x86-64...**

a. RAX es un registro salva-invocante, por eso en cualquier función hay que salvarlo antes de usarlo

-> b. R10 es un registro salva-invocante, por eso si es necesario hay que salvarlo antes de llamar a función

c. R11 es un registro salva-invocado, por eso en cualquier función hay que salvarlo antes de usarlo

d. RBP es un registro salva-invocado, por eso si es necesario hay que salvarlo antes de llamar a función

**176. Una función C llamada get\_el() genera el siguiente código ensamblador. Se puede adivinar que:**

**leaq (%rdi,%rdi,4), %rax**

**addq %rax, %rsi**

**movl arr(%rsi,4), %eax**

**ret**

a. arr es un array multi-nivel (punteros a enteros) de cuatro filas

b. arr es un array multi-nivel pero no se pueden adivinar las dimensiones

c. arr es un array bidimensional de enteros, no se pueden adivinar dimensiones

-> d. arr es un array bidimensional de enteros, con cinco columnas

**177. En un sistema Linux x86-64, ¿cuál de las siguientes expresiones es equivalente a la expresión C (x[2] + 4)[3]? Suponer que previamente se ha declarado int \*\*x.**

a)  $\ast(\ast(x + 16)) + 28$

cierto si se hubiese hecho typecast (void\*)x para evitar aritmética punteros

b)  $\ast(\ast(x + 2) + 4) + 3$

sería  $x[2][4]+3$ , la última suma sería suma entera, no aritmética punteros

-> c)  $\ast(\ast(x + 2) + 4) + 3$

sería  $x[2][4+3]$ ,  $+4+3 == +7$  sí sería aritmética punteros

d)  $\ast(\ast\ast x + 2) + 7$

## **PREGUNTAS TEST EC TEMA 3**

### **1. ¿Cómo actúa el indicador de signo?**

- a) Se pone a 1 cuando el resultado es positivo
- > b) Se pone a 1 cuando el resultado es negativo
- c) Se pone a 1 cuando el resultado es distinto
- d) Se pone a 0 cuando el resultado es negativo

### **2. La microprogramación vertical se caracteriza por tener:**

- a) escaso o ningún solapamiento entre campos
- b) capacidad para expresar un alto grado de paralelismo en las microoperaciones a ejecutar
- c) microinstrucciones largas
- > d) mucha codificación

### **3. Un procesador con una unidad de control microprogramada tiene una memoria de control de 256 palabras de 16 bits, de las que 128 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?**

- > a) 256 bits
- b) No se produce ahorro
- c) 4096 bits
- d) 3840 bits

### **4. La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...**

- > a) dos multiplexores de 2 a 1
- b) tres conexiones directas al bus común
- c) tres demultiplexores
- d) dos buffers triestado

### **5. Sea un formato de microinstrucción que incluye dos campos independientes de 9 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?**

- a) 1
- b) 4
- > c) 8
- d) 9

**6. ¿Cómo actúa el indicador N del registro de indicadores de estado?**

- > a) Se pone a 1 cuando el resultado es negativo.
- b) Se pone a 1 cuando el resultado es positivo.
- c) Se pone a 1 cuando el resultado de una operación es 0.
- d) Se pone a 0 cuando el resultado es negativo.

**7. Los procesadores comerciales con unidad de control microprogramada suelen almacenar los micropogramas...**

- a) en una RAM.
- b) en un banco de registros.
- > c) en una ROM.
- d) en una PLA.

**8. Respecto a MBR y MAR**

- a) Ambos permiten guardar información sobre el marco de pila //sin relación directa
- b) Ambos son accesibles por el programador //ninguno
- c) MAR contiene el dato/instrucción que se leerá o escribirá en memoria //ese es MBR
- > d) MAR requiere menos señales de control que MBR //en Tema 3 tr.10, sólo una (LoadMAR),

//frente a 4 de MBR (Load/Enable Mem/Bus)

**9. Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?**

- a)  $k + n \cdot 2^m$
- b)  $k + n^m$
- > c)  $k + n \cdot (2^m - 1)$
- d) Ninguno de los anteriores

**10. Un computador tiene una memoria de control de 640 palabras de 70 bits, de las que 280 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación en lugar de microprogramación?**

- >a) 19440
- b) 42280
- c) 9840
- d) ninguno de los anteriores resultados es exacto.

**11. Un computador tiene una memoria de control de 16000 palabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorramos usando nanoprogramación en lugar de microprogramación?**

- > a) 3744250
- b) 259206
- c) 287935
- d) ninguno de los resultados anteriores es exacto

**12. En un camino de datos con un solo bus, para realizar la operación de copia de un registro r1 en un registro r2, es decir  $r2 \leftarrow r1$ , es necesario:**

- a) Activar la carga del registro r1 y habilitar la salida triestado del registro r2
- b) Habilitar la salida triestado del registro r2 y activar la carga de los registros r1 y r2
- c) Habilitar las salidas triestado de los registros r1 y r2 y activar la carga del registro r2
- > d) Habilitar la salida triestado del registro r1 y activar la carga del registro r2

**13. Respecto a las unidades de control nanoprogramadas:**

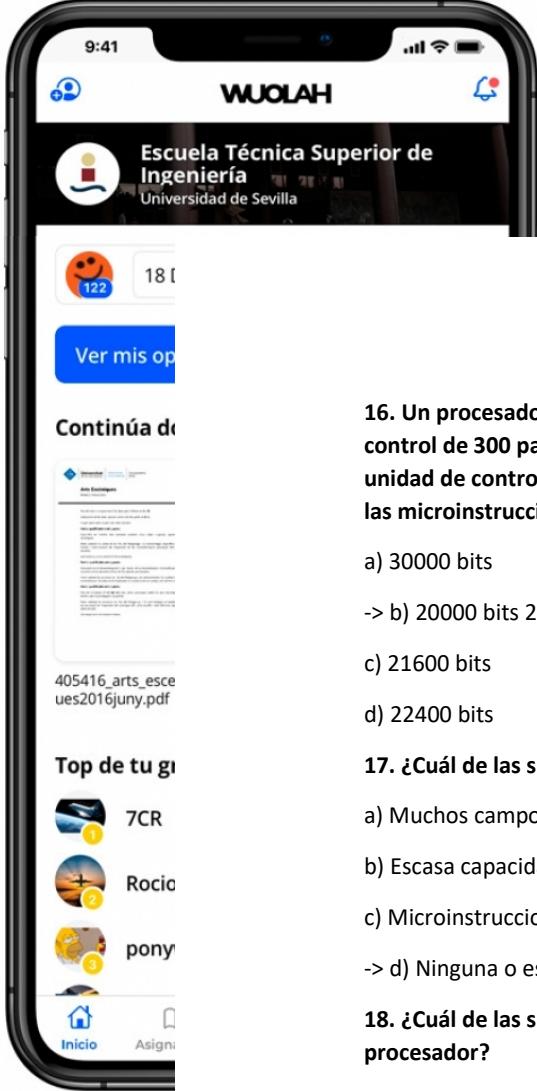
- > a) La anchura de la memoria de nanoprograma es la misma que la de memoria de microprograma en un diseño de la misma unidad de control que no usara nanoprogramación.
- b) La realización nanoprogramada de una unidad de control es más rápida que la microprogramada.
- c) El diseño de las unidades de control nanoprogramadas debe ser vertical.
- d) Suponiendo una memoria de microprograma con n microinstrucciones de w bits cada una, de las cuales  $2^m$  son distintas, el ahorro en bits si se utiliza nanoprogramación es  $(n \cdot m + 2^m \cdot w) - n \cdot w$ .

**14. En el pseudocódigo usado para representar las microinstrucciones, la expresión “goto f(IR)”:**

- a) Permite saltar a la dirección de memoria de control del principio de un microbucle.
- b) Se utiliza para realizar un microsalto condicional en función del registro de estado.
- c) Realiza una llamada a una microsubrutina.
- > d) Salta a una dirección de memoria de control que depende de la instrucción máquina actual.

**15. Para conectar las salidas de dos registros hacia un bus común en el datapath...**

- > a) se pueden usar dos buffers triestado.
- b) no se puede usar un multiplexor.
- c) se puede realizar una conexión directa.
- d) se puede usar un demultiplexor.



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**16. Un procesador con una unidad de control microprogramada tiene una memoria de control de 300 palabras de 100 bits, de las que 200 son diferentes. Si se rediseñara como unidad de control nanoprogramada, ¿qué tamaño ocuparía la nanomemoria que contiene las microinstrucciones completas sin repeticiones?**

- a) 30000 bits
- > b) 20000 bits 200 uinstr. x 100 bits
- c) 21600 bits
- d) 22400 bits

**17. ¿Cuál de las siguientes características es típica de la microprogramación horizontal?**

- a) Muchos campos solapados.
- b) Escasa capacidad para expresar paralelismo entre microoperaciones.
- c) Microinstrucciones cortas.
- > d) Ninguna o escasa codificación.

**18. ¿Cuál de las siguientes funciones no corresponde a la unidad de control de un procesador?**

- > a) Calculo de operaciones de coma flotante
- b) Decodificación de las instrucciones
- c) Secuenciamiento de las instrucciones
- d) Generación de las señales de control que provocan la ejecución de cada instrucción

**19. Para el procesador con unidad de control microprogramada estudiado en clase, Tanenbaum define 15 codops de 4bits para instrucciones con un operando dirección de 12bits (LODD M, STOD M, ADDD M, ...), y en lugar de gastar el último codop 1111 en otra instrucción con una dirección, realiza una extensión de codop de 3bits para definir ocho instrucciones más. En general esas instrucciones no tienen ningún operando, salvo INSP Y (sp+=Y) y DESP Y (sp-=Y), que tienen un operando inmediato.**

a) Y es un operando de 8 bits y no sobra espacio de codificación (huecos) para modificaciones sin aprovechar bit8

-> b) Y es un valor de 8 bits pero podría definirse que fuera de 9 bits sin ninguna dificultad

c) Y es una dirección corta de 9 bits y podría definirse que fuera de 10 u 11 bits cambiando la codificación por extensión

// 11bits imposible, sólo cabrían INSP/DESP, no habría sitio para PUSH/POP etc

d) Y ocupa 10 bits, pero podría ser de 11 bits cambiando la codificación por extensión y uniendo ambas instrucciones en una sola ADDSP Y (sp+=Y) en donde Y se interpretara como número con signo en complemento a dos

// Y no ocupa 10bits.

//Sí que sería posible ADDSP codop 1111 1yyy yyyy yyyy con 11bit para Y, y entonces codop //1111 0ccc 0000 0000 permite de sobra los 6 codops restantes. [T3.4Tnbaum] tr.79

**20. Una unidad de control microprogramada con secuenciamiento explícito con dos direcciones por microinstrucción, tiene una memoria de control con 35 bits de longitud de palabra. Si las microinstrucciones emplean 15 bits en total para los campos de control y de tipo y condición de salto, el número máximo de palabras de la memoria de control de esta unidad de control microprogramada es de:**

- a) 10
- b)  $2^{20}$
- > c)  $2^{10}$
- d) 20

**21. ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?**

- > a) Una memoria.
- b) Un registro.
- c) Un multiplexor.
- d) Un contador.

**22. En una arquitectura RISC típica:**

- a) no puede usarse segmentación.
- b) la programación resulta mucho más simple que en una arquitectura CISC.
- > c) se usa un porcentaje elevado de las instrucciones del repertorio.
- d) la UC es más compleja que en una arquitectura CISC.

//Evitar esta pregunta (de la antigua titulación), las transparencias que hablan de RISC están //ahora repartidas por los distintos temas

**23. Sea un formato de microinstrucción que incluye dos campos independientes de 10 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?**

- a) 14
- b) 13
- c) 10
- >d) 9

**24. La salida de un campo del registro de microinstrucción que solapa dirección de salto y algunas señales de control han de conectarse a:**

- a) una ROM o PLA
- >b) un demultiplexor controlado por el tipo de salto
- c) el registro de instrucción
- d) la memoria de control

**25. Una unidad de control microprogramada se denomina "con secuenciamiento de microinstrucciones explícito" según tenga o no tenga**

- a) ROM/PLA para traducir el codop en dirección de inicio de microprograma (goto f(IR))  
//todos los diseños implícitos en las transparencias usan ROM traducción, pero también se  
//podría añadir ROM al explícito
- b) microcódigo de decodificación que analice el codop bit a bit de izquierda a derecha  
//con ROM traductora (goto f(IR)) no haría falta ese análisis
- > c) micro-contador de programa atacando a las líneas de dirección de la memoria de control  
// si y sólo si tiene micro-PC, sería implícito
- d) un multiplexor para seleccionar la fuente de la dirección de la memoria de control  
//todos los diseños en las transparencias tienen un MUX en la dirección de la memoria de  
//control (o en el micro-PC)

**26. ¿Cuál de las siguientes afirmaciones es verdadera?**

- > a) La unidad de control necesita como entrada el registro de estado para poder controlar la ejecución de las instrucciones de salto condicional.
- b) El registro puntero de pila es un registro de propósito general que suele contener tanto direcciones como datos.
- c) Las únicas instrucciones en las que algunas de sus fases de ejecución llevan un acceso a memoria son las instrucciones load y store.
- d) El registro de instrucción es un registro de propósito específico que contiene la dirección de la siguiente instrucción a ejecutar.

**27. En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de**

- a) 17 bits
- b) 16 bits
- > c) 5 bits
- d) 4 bits

**28. Para realizar la microoperación MAR <- PC, habrá que activar:**

- a) EnPC y EnMAR
- b) LdPC y EnMAR
- > c) EnPC y LdMAR
- d) LdPC y LdMAR

**29. Una posible codificación en microinstrucciones de la instrucción CALL X es:**

- a) SP=SP-1 ; m[SP]=PC ; PC=PC+1
- > b) SP=SP-1 ; m[SP]=PC ; PC=X
- c) PC=X ; SP=SP-1 ; m[SP]=PC
- d) SP=PC-1 ; m[SP]=PC ; PC=X

**30. Dado un camino de datos concreto, un posible formato de microprogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)**

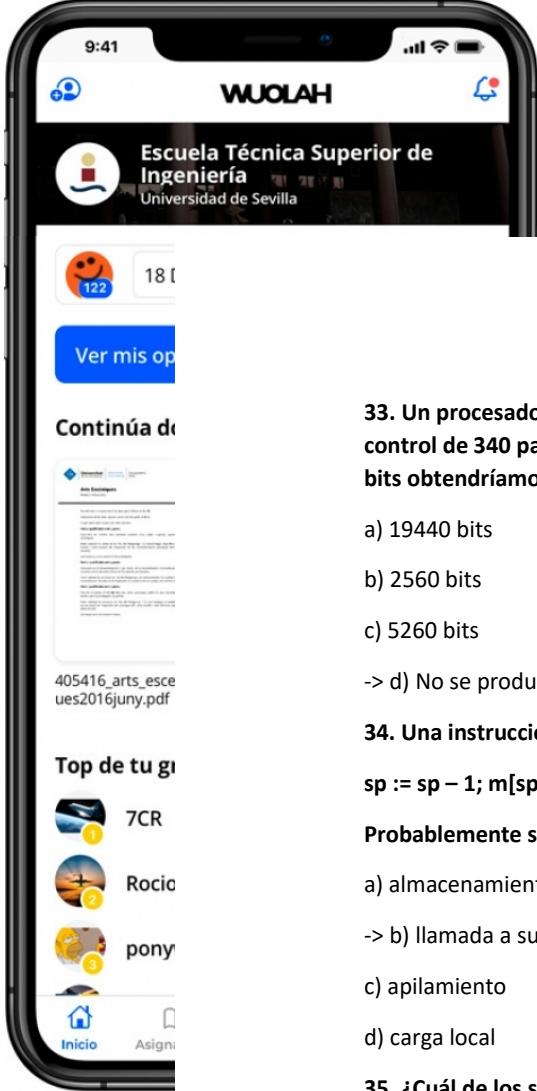
- a) longitud relativa de microinstrucción
- > b) microbifurcaciones
- c) solapamiento
- d) codificación

**31. Alguna de las siguientes \*no\* es una operación básica de la unidad de control:**

- a) realizar operación ALU y guardar resultado en registro
- b) transferir un registro a otro
- > c) (guardar o recuperar) un registro (en / de) la pila
- d) (leer o escribir) un registro (de / a) memoria

**32. Un computador tiene una memoria de control de 16 Kpalabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorramos usando nanoprogramación en lugar de microprogramación?**

- a) 3928652
- b) 259206
- c) 287935
- > d) ninguno de los resultados anteriores es exacto



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**33. Un procesador con una unidad de control microprogramada tiene una memoria de control de 340 palabras de 16 bits, de las que 180 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?**

a) 19440 bits

b) 2560 bits

c) 5260 bits

-> d) No se produce ahorro

**34. Una instrucción máquina puede desglosarse en las siguientes operaciones elementales:**

$sp := sp - 1; m[sp] := pc; pc := x$

Probablemente se trate de una instrucción de:

a) almacenamiento local

-> b) llamada a subrutina

c) apilamiento

d) carga local

**35. ¿Cuál de los siguientes grupos de señales son entradas a la unidad de control?**

a) Las señales de carga/incremento/desplazamiento de registros

b) Las señales de selección de entradas de multiplexores del datapath

-> c) Los bits del registro de indicadores (flags)

d) Los bits de las opciones b y c

**36. Respecto a los términos microinstrucción y microcódigo:**

a) Son equivalentes, llamamos microcódigo o microinstrucción a una palabra de la memoria de control

b) Una microinstrucción está programada en microcódigo, que es un lenguaje para programar señales de control

c) Un microcódigo controla una serie de señales de control relacionadas (por ejemplo, el código 000 para que la ALU realice la suma), y varios juntos forman una microinstrucción

-> d) Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoria

**37. El control residual se utiliza para:**

a) reducir el tiempo de ejecución de las instrucciones máquina

b) eliminar los bits residuales de la ejecución de los microinstrucciones

-> c) reducir el tamaño de la memoria de control

d) ninguna de las anteriores es cierta

**38. Son funciones de la unidad de control:**

- a) la codificación de las instrucciones máquina
- b) la lectura de memoria principal de la instrucción apuntada por el µPC
- > c) el secuenciamiento de las instrucciones máquina
- d) todas las respuestas son ciertas

**39. En cuanto al control microprogramado:**

- a) se guardan en una ROM las instrucciones máquina del conjunto de instrucciones.
- b) la UC se construye con puertas lógicas, biestables, etc.
- > c) se usa en CISC para facilitar el diseño de la UC tan compleja.
- d) la lentitud en la ejecución de las instrucciones máquina la impone directamente la tecnología hardware usada.

**40. Alguna de las siguientes señales no sirve de entrada a la unidad de control. ¿Cuál?**

- a) señal de reloj (CLK)
- b) instrucción actual (bits del registro IR)
- c) estado de la unidad de proceso (flags Z, S, C, O...)
- > d) contador de programa (bits del registro PC)

**41. [T3.1] ¿Cuál de las siguientes funciones es una tarea propia de la unidad de control en la CPU?**

- > a) decodificar las instrucciones del programa
- b) almacenar instrucciones del programa
- c) realizar operaciones lógicas
- d) almacenar datos del programa

**42. En una arquitectura RISC típica:**

- a) se usan pocas instrucciones de las disponibles en el conjunto de instrucciones
- > b) suele usarse segmentación
- c) la programación resulta mucho más simple que en una arquitectura CISC
- d) la UC es más compleja que en una arquitectura CISC

**43. ¿Cómo actúa el indicador Z del registro de indicadores de estado?**

- a) Se pone a 1 cuando el resultado es negativo
- b) Se pone a 0 cuando el resultado es negativo
- > c) Se pone a 1 cuando el resultado de una operación es 0
- d) Se pone a 1 cuando el resultado es positivo

**44. Alguna de las siguientes señales no es salida de la unidad de control. ¿Cuál?**

- a) señales de lectura y escritura en memoria (RD, WR)
- > b) dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)
- c) códigos de selección en multiplexores, decodificadores, ALU, etc (00, 01, 10, 11...)
- d) señales de carga, habilitación y/o desplazamiento de registros (Load, Enable, ShiftL, ShiftR)

**45. Sea un formato de microinstrucción que incluye dos campos independientes de 8 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?**

- a) 1
- b) 8
- c) 9
- > d) 7

# TEST EC TEMA 4

**1. Cuando dos o más instrucciones necesitan un recurso hardware en el mismo ciclo, se trata de un riesgo:**

- > a) estructural
- b) de control
- c) por dependencia de datos
- d) de salto

**2. La ganancia en velocidad ideal de un cauce de K etapas de igual duración T ejecutando un programa de N instrucciones es**

- > a)  $S = KN/(K+N-1)$
- b)  $S = KN/(K-N+1)$
- c)  $S = NT/(N+K-1)T$
- d)  $S = NKT/(N-K+1)T$

**3. Un procesador de 1GHz tarda 4ns en realizar 4 instrucciones sin realizar segmentación de cauce. ¿Cuánto tardaría en realizar 9 instrucciones una versión de dicho procesador con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones?**

- a) 2 ns
- b) 9 ns
- > c) 3 ns

1ns para que acabe la 1<sup>a</sup> instrucción, y las restantes 8 van saliendo a cada 1ns/4etapas, o sea que duran otros 2ns

- d) 4.5 ns

**4. Tipos de riesgos que hemos estudiado en cauces segmentados (señalar la opción incorrecta)**

- a) riesgos de control
- b) riesgos estructurales
- > c) riesgos de transferencia
- d) riesgos de (dependencia de) datos

**5. Respecto a la predicción de saltos, alguna de las siguientes afirmaciones es falsa**

- a) si se toma la misma decisión para cada tipo de instrucción, se trata de "predicción estática"
- b) si la predicción cambia según la historia de ejecución del programa, se trata de "predicción dinámica"
- > c) para predicción estática, es conveniente decidir que los saltos hacia adelante siempre se cumplen, y hacia atrás no  
en los bucles se fallaría siempre (salvo la última iteración)
- d) para predicción dinámica, existen, entre otros, algoritmos de dos o cuatro estados, que requieren 1 o 2 bits por instrucción

**6. Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:**

- a) 80
- b) 5
- c) 2,5
- > d) 4

**7. Una cola de precaptación sirve para:**

- a) Disminuir el periodo de ciclo del cauce
- b) Resolver ciertos problemas de dependencia de datos
- c) Aumentar el número de etapas del cauce
- > d) Reducir el efecto de los fallos de cache

**8. Si representamos la fase Decode con una D, Execute con una E, Fetch con una F y Writeback con una W, el orden correcto de las distintas fases de una instrucción máquina es:**

- a) F W D E
- b) D E F W
- c) D F E W
- > d) F D E W

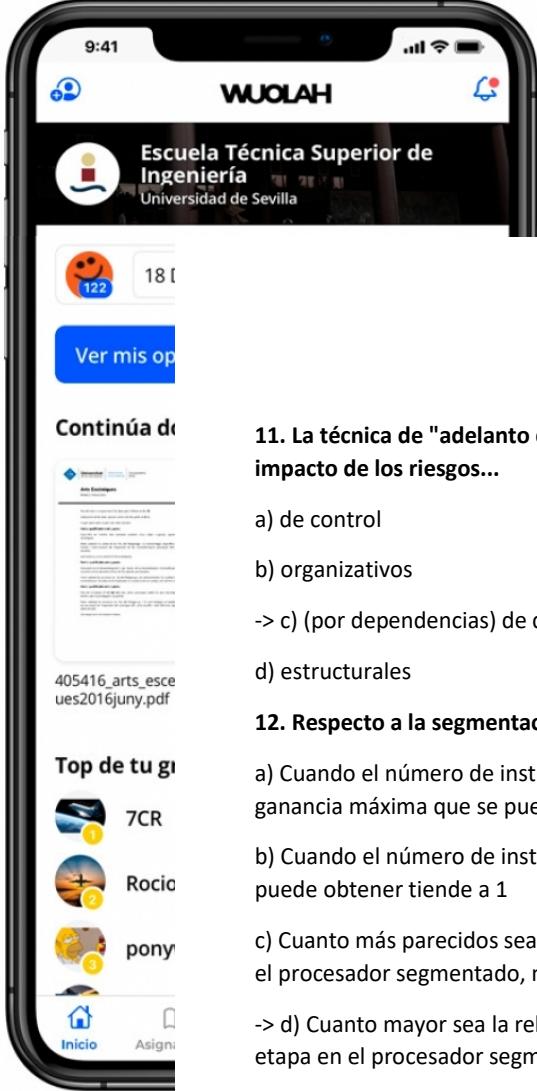
**9. En la secuencia de instrucciones siguiente, siendo el primer registro el destino, ¿cuántos riesgos se dan?**

sub r2,r1,r3  
or r8,r6,r2

- > a) Un riesgo por dependencia de datos  
la resta calculada en r2 debe estar disponible para el or con r6
- b) Dos riesgos por dependencia de datos y uno de control
- c) Un riesgo estructural
- d) Un riesgo estructural y dos por dependencia de datos

**10. Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para escribir el resultado en el banco de registros, ¿cuál es la frecuencia de reloj máxima del procesador?**

- > a) 66,67 MHz
- b) 500 MHz
- c) 200 MHz
- d) 40 MHz



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the App Store

GET IT ON Google Play

**11. La técnica de "adelanto de registros" (register forwarding) en un cauce segmentado se usa para limitar el impacto de los riesgos...**

- a) de control
- b) organizativos
- > c) (por dependencias) de datos
- d) estructurales

**12. Respecto a la segmentación:**

- a) Cuando el número de instrucciones ejecutadas tiende al número de etapas de un procesador segmentado, la ganancia máxima que se puede obtener tiende a infinito
- b) Cuando el número de instrucciones ejecutadas en un procesador segmentado crece, la ganancia máxima que se puede obtener tiende a 1
- c) Cuanto más parecidos sean el tiempo de ejecución de una instrucción sin segmentar y el tiempo de una etapa en el procesador segmentado, mayor será la ganancia máxima que se puede obtener
- > d) Cuanto mayor sea la relación entre el tiempo de ejecución de una instrucción sin segmentar y el tiempo de una etapa en el procesador segmentado, mayor será la ganancia máxima que se puede obtener

**13. Respecto a la segmentación, ¿cuál de las siguientes afirmaciones es falsa?**

- > a) Retrasar la fase de decisión saltar/no saltar de las instrucciones de salto condicional contribuye a mejorar el rendimiento del procesador
- b) La técnica de register forwarding habilita una serie de caminos (buses) que se añaden al cauce para permitir que los resultados de una etapa pasen como entradas a la etapa donde son necesarias
- c) Cuantas más etapas tenga un cauce, más instrucciones se estarán ejecutando en distintas fases y más posibilidades se presentan de que existan riesgos entre ellas
- d) La reorganización del código y la introducción de instrucciones no permite evitar dependencias de datos

**15. Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se aproxima a:**

- a) 0,25
- b) 5
- c) 20
- > d) 4

**16. Un sistema no segmentado tarda 200 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce segmentado de 20 etapas con un ciclo de reloj de 12 ns. Cuando se procesan muchas instrucciones, la máxima ganancia de velocidad que podría obtenerse se acerca a:**

-> a) 16,67

b) 1,2

c) 20

d) 1,667

**17. Los riesgos de datos consisten en que...**

a) dos instrucciones acceden a la vez al mismo dato

-> b) una instrucción necesita un dato calculado por otra anterior

c) dos instrucciones necesitan leer el mismo dato

d) todas las respuestas anteriores son correctas

**18. Las instrucciones de salto...**

a) son las causantes de los riesgos de tipo RAW y WAW.

-> b) complican el diseño eficiente de los procesadores segmentados.

c) siempre utilizan direccionamiento absoluto.

d) Todas las afirmaciones anteriores son ciertas.

**19. Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 6 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se aproxima a:**

a) 4

b) 5

-> c) 3,33

d) 0,3

**20. La precaptación (cola de instrucciones) está relacionada con...**

a) Los riesgos de control (intenta determinar de antemano el flujo de control)

-> b) Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)

c) Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)

d) Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)

**21. Alguno de los siguientes NO es un motivo de que no se alcance la ganancia ideal en un cauce segmentado**

-> a) La emisión múltiple (y posiblemente desordenada) de instrucciones

T4 tr.19

b) El propio coste de la segmentación (carga de los registros de acople, etc...)

c) Los riesgos (hazards)

d) La duración del ciclo de reloj impuesta por la etapa más lenta

**22. Respecto a los conceptos de procesamiento segmentado y superescalar, una de las siguientes afirmaciones es falsa**

-> a) en cualquier procesador resulta ventajoso usar una cola de instrucciones, pero es más importante para uno segmentado (fundamental) que para uno superescalar (conveniente)

## Tema 4 tr.46

- b) implícitamente, se presupone que un procesador superescalar emitirá más de una instrucción por ciclo
- c) idealmente, con el segmentado se intenta ejecutar una instrucción por ciclo, y con el superescalar más de una por ciclo (al combinarlo con segmentado)
- d) por definición, un procesador superescalar debe tener varias unidades funcionales (más de una)

### 23. ¿Cuál de las siguientes afirmaciones es cierta?

- a) Al realizar la segmentación de cauce aumenta en general el tiempo necesario para la ejecución de un programa
  - b) Debido a que pueden existir dependencia de datos, los resultados de un programa pueden ser diferentes a si el programa se ejecutara sin segmentación
  - c) La segmentación de cauce disminuye el número de instrucciones necesarias para la ejecución de un programa
- > d) Ninguna de las afirmaciones anteriores

### 24. En la técnica de salto retardado:

- a) El salto se realiza varios ciclos antes de la instrucción de salto
- > b) El compilador puede reorganizar el código para llenar los huecos de retardo con instrucciones útiles
- c) Las instrucciones en los huecos de retardo se ejecutan unas veces y otras no
- d) El compilador no puede insertar operaciones NOP en los huecos de retardo

### 25. La predicción de saltos está relacionada con...

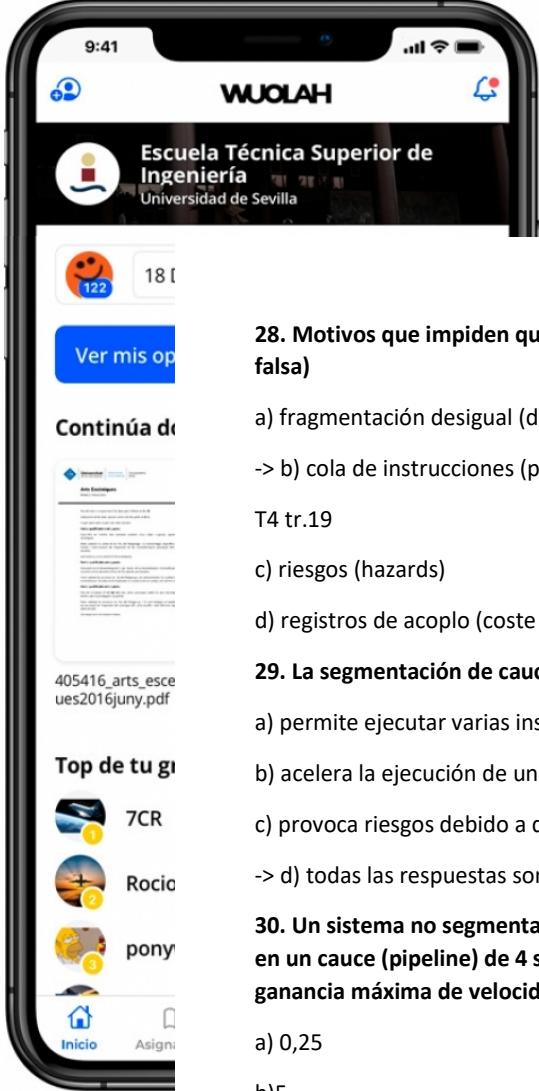
- a) Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
  - b) Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)
  - c) Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
- > d) Los riesgos de control (intenta determinar de antemano el flujo de control)

### 26. Un procesador está segmentado en las etapas F, D, E, M y W. Cada una de ellas consume un tiempo t. La aceleración ideal (si no hay riesgos) al ejecutar n instrucciones respecto a un procesador no segmentado será:

- > a)  $5n / (4+n)$
- kn/(k+n-1), con k=5
- b)  $(5+n) / 4t$
- c)  $(4+n) / 5t$
- d)  $4n / (5+n)$

### 27. Un sistema no segmentado tarda 10 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 5 segmentos con un ciclo de reloj de 4 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:

- a) 50
- b) 5
- > c) 2,5
- d) 4



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**28. Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señalar la respuesta falsa)**

- a) fragmentación desigual (duración desigual de etapas)
- > b) cola de instrucciones (precaptación)
- T4 tr.19
- c) riesgos (hazards)
- d) registros de acople (coste de la segmentación)

**29. La segmentación de cauce...**

- a) permite ejecutar varias instrucciones concurrentemente
- b) acelera la ejecución de un programa
- c) provoca riesgos debido a datos
- > d) todas las respuestas son ciertas

**30. Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se acerca a:**

- a) 0,25
- b) 5
- c) 20
- > d) 4

**31. ¿Cuál de las siguientes afirmaciones sobre la segmentación de cauce es cierta?**

- a) El CPI de un cauce superescalar es siempre 1 o menor que 1
- > b) En general, una operación segmentada ("pipelined") requiere el mismo tiempo o más, desde el principio hasta el fin, que la misma operación en una implementación no segmentada
- c) La predicción de saltos es una técnica para minimizar los riesgos de datos
- d) Un cauce ("pipeline") de instrucciones inicialmente vacío y con 3 etapas tardará siempre 5 ciclos de reloj en ejecutar 3 instrucciones si cada una de ellas utiliza las 3 etapas

**32. ¿Cuál de los siguientes modos de direccionamiento es menos preferible para un procesador de 32 bits y con tamaño de instrucción de 32 bits?**

- a) indexado
- b) registro
- c) indirecto a través de registro
- > d) directo (o absoluto)

Los otros 3 están indicados como preferibles en Tema 4 tr.42. Si una instrucción ocupa 32bits, no hay sitio ni para el codop si la propia dirección (del argumento con direccionamiento directo) ocupa de por sí los 32bits.

**33. Si el proceso de empaquetado de un producto (50 segundos de duración) puede segmentarse en 5 etapas, cada una de 10 segundos, de modo que 5 operarios puedan trabajar cada uno en una etapa, ¿cuál de las siguientes afirmaciones es falsa al aplicar la segmentación?**

- a) La preparación completa de cada producto sigue requiriendo 50 s (igual que con una sola persona empaquetando los productos)
- > b) Cada 50 s saldrá un nuevo producto empaquetado, el mismo tiempo que cuando no había cadena de empaquetamiento
- c) La ganancia de velocidad (aceleración) con la segmentación es de 5
- d) Una vez en funcionamiento la segmentación, se tardará 100 segundos en tener 10 productos empaquetados, mientras que con un solo operario se tardaría 500 segundos.

**34. Sobre la segmentación:**

- > a) Existen limitaciones al rendimiento provocadas por las instrucciones de salto y por las dependencias de datos.
- b) La frecuencia de reloj viene impuesta por la etapa más corta.
- c) Es una técnica para comenzar simultáneamente la ejecución de varias instrucciones con el fin de reducir el tiempo de ejecución.
- d) Un procesador superescalar no puede estar segmentado.

**35. Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?**

- a) el retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca
- b) el retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple
- > c) el retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de salto

T4 tr.38

- d) el retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)

**36. ¿Cuál de los siguientes modos de direccionamiento es \*menos\* preferible para un procesador con segmentación de cauce?**

- > a) Indirecto a través de memoria
- b) Indirecto a través de registro
- c) Registro
- d) Indexado (o relativo a base, o base+índice)

**37. En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:**

- > a) Un incremento de las prestaciones
- b) Un mayor retraso en la ejecución de los programas debido al incremento del número de etapas
- c) Una disminución de la máxima frecuencia de reloj a la que puede operar el cauce
- d) Una disminución en la posible dependencia de datos

**38. Un sistema no segmentado tarda 20 ns en procesar una instrucción. Las instrucciones pueden ser procesadas en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 6 ns. Cuando se procesan muchas instrucciones, la ganancia máxima de velocidad que se obtiene se aproxima a:**

- a) 4
- b) 0,3
- c) 5
- > d) 3,33

**39. Un procesador está segmentado en cinco etapas. Cada una de ellas consume un tiempo t. La aceleración ideal (si no hay riesgos) al ejecutar n instrucciones respecto a un procesador no segmentado será:**

- > a.  $5n / (4+n)$  //T4 tr.18,  $kn/(k+n-1)$ , con k=5
- b.  $(4+n) / 5t$
- c.  $4n / (5+n)$
- d.  $(5+n) / 4t$

# TEST TEMA 5 EC

**1. ¿A qué tipo de interrupciones corresponde la forma de determinar la dirección de comienzo de una rutina de interrupción en la que se envía una instrucción de bifurcación completa?**

- a) Interrupciones no vectorizadas
- b) Interrupciones con direcciones fijas
- > c) Interrupciones vectorizadas
- d) Interrupciones encadenadas

**2. Si se dice que en un sistema computador cada dirección especifica uno o dos puertos de E/S, se refiere a que:**

- a) La misma dirección (por ejemplo 0x0210) puede ser una posición de memoria o un puerto de E/S, según IO/M#
- > b) Un puerto será de sólo lectura, otro de sólo escritura, y ambos se decodifican en la misma dirección
- c) La misma dirección puede usarse para transferir un byte o una palabra de mayor tamaño (ese byte y el siguiente)
- d) La pregunta es capciosa, una dirección puede especificar un puerto, no dos

**3. ¿Qué técnica de E/S se dice controlada por hardware?**

- a) E/S programada
- b) E/S controlada por interrupciones
- > c) Acceso directo a memoria
- d) Ninguna de las anteriores

**4. Se dispone de un procesador con una frecuencia de reloj de 1 GHz. Se le conecta un dispositivo que genera 100.000 interrupciones por segundo. La rutina de servicio de interrupción ejecuta 500 instrucciones. El número medio de ciclos por instrucción es 2. ¿Qué porcentaje del tiempo dedica el procesador al dispositivo?**

- a) 90%
- > b) 10%
- c) 50%
- d) 1%

**5. Señale cuál de las siguientes opciones es una técnica habitual para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:**

- a) E/S por nivel
- > b) Acceso directo a memoria (DMA)
- c) E/S por flanco
- d) Acceso indirecto a memoria (IMA)

**6. El fragmento de código ensamblador de un microprocesador de 8 bits**

**lds IOBuf ; Apuntar puntero pila a**

**; ...área mem.intermedia**

**Idx Count ; Inicializar X-contador**

**poll Ida a Status; Leer estado en A**

**bpl poll ; Signo(A)!=1 => repetir**

**Ida a Data ; Leer dato en A**

**psh a ; Transferir dato a pila**

**dex ; Decrementar contador X**

**bne poll ; Seguir leyendo si X!=0**

**corresponde a:**

-> a) Entrada programada con consulta de estado

Ida a Data -> lectura del puerto de Datos

Ida a Status -> poll (consulta) de estado

b) Entrada programada sin consulta de estado

c) Salida programada sin consulta de estado

d) Salida programada con consulta de estado

**7. La instrucción máquina DI (Disable Interrupts), conocida como CLI (Clear Interrupt Flag) en x86, se utiliza para desactivar:**

-> a) Todas las interrupciones enmascarables

Tema 5 tr.50

b) Determinados niveles de interrupción de forma selectiva

c) Las interrupciones de inferior o igual prioridad a una dada

d) Las interrupciones software

**8. [T5.3] Técnicas que se pueden usar para determinar la causa de una interrupción (señalar la opción incorrecta)**

- a) interrupciones vectorizadas
- > b) línea de reconocimiento INTA#
- c) consulta de estado, o polling
- d) múltiples líneas de interrupción INT1#, INT2#...

**9. [T5.1] Señale cuál de las siguientes opciones no es un modo para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:**

- > a) E/S por flanco
- b) E/S iniciada por interrupción
- c) Acceso directo a memoria (DMA)
- d) E/S programada

**10. ¿Cuál de las siguientes tareas no es responsabilidad de un circuito de interfaz o controlador de periféricos sencillo?**

- > a) Ejecutar el programa de transferencia de información entre el procesador y los dispositivos de E/S
- b) Adaptar el formato de las señales
- c) Recibir señales de control desde el procesador
- d) Ajustar la temporización entre el procesador y los dispositivos de E/S

**11. Las interrupciones generadas por el teclado interrumpirán al procesador:**

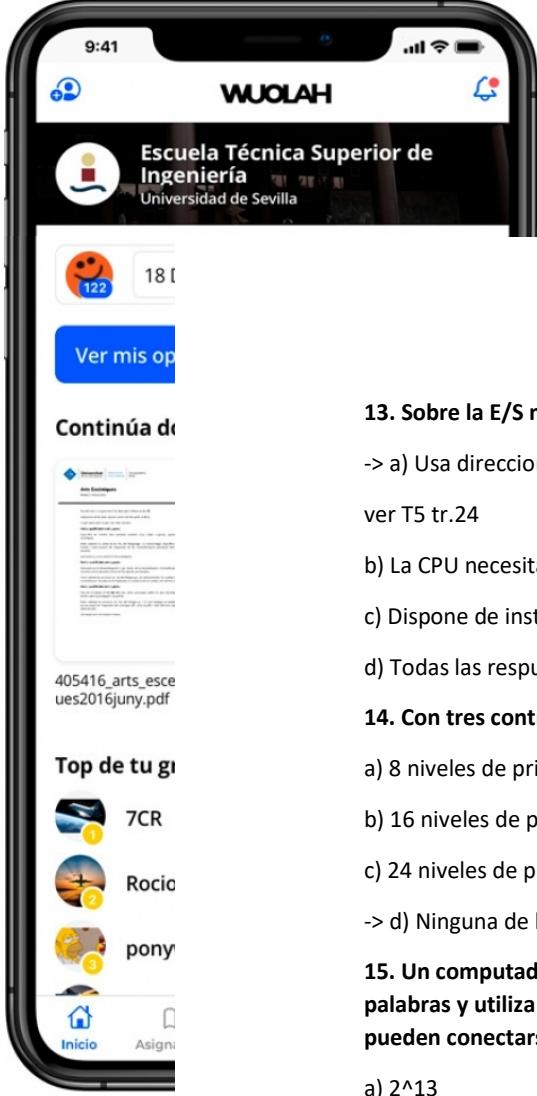
- a) sólo cuando el procesador no esté realizando un trabajo útil
- b) sólo si el procesador está chequeando el estado del teclado
- > c) sólo si el procesador tiene activado el indicador de habilitación de interrupciones
- d) siempre que el usuario pulse una tecla en el teclado

**12. ¿Cuántos niveles de interrupción podremos gestionar si disponemos de 7 controladores de interrupciones programables 8259?**

- > a) 50

ver T5 tr.77: 1 master + 6 slaves =  $2 + (8 \times 6) = 50$  IRQ

- b) 45
- c) 48
- d) 56



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**13. Sobre la E/S mapeada en memoria podemos decir que:**

-> a) Usa direcciones de memoria para acceder a puertos de E/S

ver T5 tr.24

b) La CPU necesita el pin IO/M#

c) Dispone de instrucciones especiales de E/S

d) Todas las respuestas anteriores son falsas

**14. Con tres controladores de interrupciones 8259 se pueden manejar exactamente:**

a) 8 niveles de prioridad

b) 16 niveles de prioridad

c) 24 niveles de prioridad

-> d) Ninguna de las anteriores es cierta

**15. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de  $2^{13}$  palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?**

a)  $2^{13}$

b)  $2^{11}$

c)  $2^{12}$

-> d)  $2^{10}$

**16. ¿Cuál de las siguientes funciones no corresponde a un controlador (interfaz) de E/S?**

a) Almacenamiento temporal de datos

b) Comunicación con el dispositivo

-> c) Almacenamiento de programas

d) Comunicación con el microprocesador

**17. ¿Qué técnica de E/S requiere menos atención por parte del procesador?**

a) Todas requieren la misma atención

-> b) E/S mediante acceso directo a memoria

c) E/S mediante interrupciones

d) E/S programada

**18. Supongamos dos procesadores con bus de direcciones con idéntico número de líneas. Si uno de ellos emplea E/S mapeada en memoria y el otro E/S independiente, ¿cuál podrá acceder a una mayor cantidad de memoria?**

- > a) El que tiene E/S independiente
- b) Depende del tamaño del bus de direcciones
- c) El que tiene E/S mapeada en memoria
- d) Ambos podrán acceder a la misma cantidad de memoria

**19. ¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?**

- a) En las transferencias síncronas
- > b) En las transferencias asíncronas
- c) En ambas
- d) En ninguna

**20. Respecto a los conceptos de procesador de E/S, canal de E/S, dispositivos de E/S:**

- > a) Un procesador o canal tiene un repertorio de instrucciones específico para manejar los dispositivos E/S
- b) Cada canal es una línea de comunicación entre el procesador y un dispositivo de E/S.
- c) Al conjunto de conexiones entre el procesador y los dispositivos se le denomina canal de E/S (de ese ordenador)
- d) La pregunta es capciosa, el procesador no es E/S, son otros dos componentes von Neumann distintos (ALU+UC)

**21. La E/S programada:**

a) Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU tiene el control de toda la operación.

b) Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque una instrucción de transferencia individual de datos con la interfaz del periférico (por ej. IN, OUT) es más lenta en E/S programada que en E/S por interrupciones.

c) Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU es más rápida que el controlador de interrupciones y la interfaz del periférico.

-> d) Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU debe encargarse de la sincronización con la interfaz del periférico haciendo una espera activa.

**22. Alguna de las siguientes técnicas NO es de utilidad para determinar la causa de una interrupción**

- a) Múltiples líneas de interrupción INT1#, INT2#...
- b) Consulta de estado, o polling
- > c) Línea de reconocimiento INTA#
- d) Interrupciones vectorizadas

**23. ¿En qué método para determinar la dirección de comienzo de una rutina de servicio de interrupción se envía parte de dicha dirección?**

- > a) Direccionamiento relativo
- b) Direccionamiento absoluto
- c) Direccionamiento indirecto
- d) Direccionamiento mediante instrucción de bifurcación

**24. ¿Es posible utilizar 4 GB de memoria en un sistema cuya CPU emplea E/S mapeada en memoria, cuyo bus de direcciones es de 32 bits y que tiene al menos un puerto de E/S? Supondremos que no se puede emplear ninguna técnica de extensión del bus de direcciones.**

- a) Sí
- > b) No
- c) Depende de si el número puertos de E/S es muy elevado
- d) Ninguna de las respuestas anteriores es correcta

**25. Respecto a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:**

- > a) La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo
- b) El puerto, o interfaz, contiene los controladores necesarios para comunicar el dispositivo con el procesador
- c) Cada puerto o interfaz es una línea de comunicación con el procesador. El conjunto de ellos forma el controlador.
- d) El interfaz contiene las controladoras necesarias para conectar los puertos con el procesador

**26. ¿En qué técnica para determinar la dirección de comienzo de la rutina de servicio de interrupción se fija dicha dirección en los circuitos de la CPU?**

- a) Envío de instrucción de bifurcación completa.
- b) Direccionamiento absoluto.
- c) Direccionamiento relativo.
- > d) Direcciones fijas.

**27. ¿De cuántos canales de E/S independientes dispone el controlador de acceso directo a memoria 8237?**

-> a) 4

ver T5 tr.91

b) 8

c) 2

d) 16

**28. La consulta de estado que se puede llevar a cabo en la E/S programada sirve para...**

-> a) consultar si el dispositivo está dispuesto para recibir datos o tiene datos disponibles

b) consultar qué dispositivo ha solicitado la interrupción

c) consultar si el dispositivo funciona correctamente

d) Ninguna de las respuestas anteriores es correcta

**29. ¿Qué técnica de E/S consume menos tiempo del procesador?**

a) E/S programada

b) E/S mediante interrupciones

-> c) E/S mediante DMA

d) Todas pueden consumir el mismo número de recursos en función de la velocidad del dispositivo de E/S al que se acceda

**30. Con nueve controladores de interrupciones 8259 se pueden manejar exactamente:**

a) 8 niveles de prioridad

b) 16 niveles de prioridad

c) 24 niveles de prioridad

-> d) Ninguna de las anteriores es cierta

En cascada, al 8259 maestro se conectan los otros 8, cada uno con 8 IRQs => 64 niveles

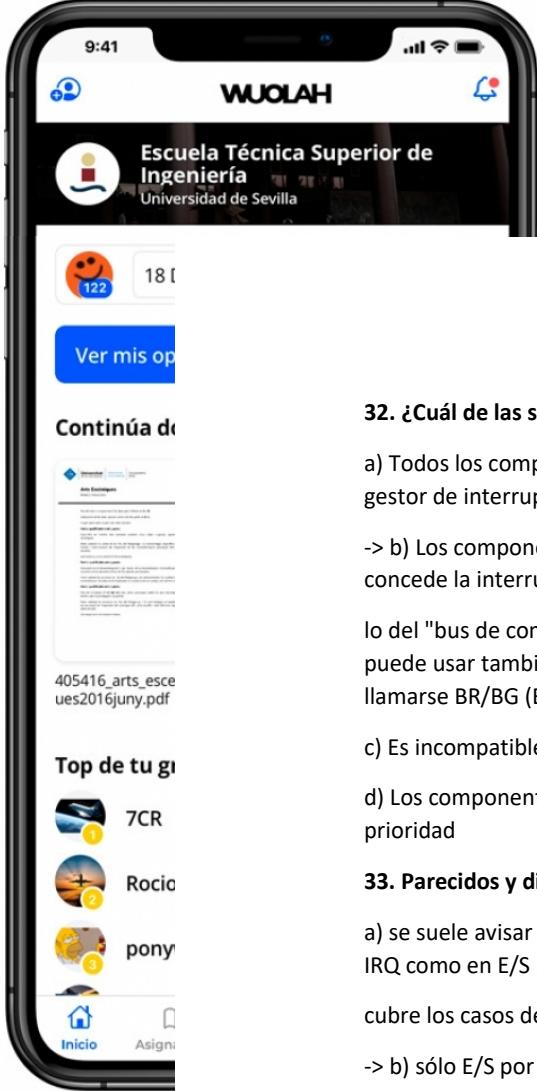
**31. ¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC (modo real) es cierta?**

a) Existen 1024 vectores de interrupción.

b) Cada vector de interrupción es una doble palabra de 32 bits formada en primer lugar (dirección menor) por el segmento y seguida por el desplazamiento (dirección mayor) de cada rutina de servicio de interrupción

c) No todas las interrupciones se pueden generar por software

-> d) Ninguna de las anteriores afirmaciones es cierta



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

### 32. ¿Cuál de las siguientes afirmaciones acerca del daisy-chain es cierta?

- a) Todos los componentes se conectan directamente y con igual prioridad al procesador o gestor de interrupciones
- > b) Los componentes se comportan de forma cooperativa: sólo al de mayor prioridad se le concede la interrupción o se apodera del bus de comunicaciones
- lo del "bus de comunicaciones" no se explica en las transparencias actuales - el daisy-chain se puede usar también como método de arbitraje para conceder el bus - las señales suelen llamarse BR/BG (Bus request/grant) en lugar de INTR/INTA (IRQ request/acknowledge)
- c) Es incompatible con la técnica de sondeo o polling
- d) Los componentes están conectados todos con todos y un gestor centralizado decide la prioridad

### 33. Parecidos y diferencias entre los métodos de E/S (señalar la opción incorrecta)

- a) se suele avisar a la CPU (con una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ como en E/S por DMA
- cubre los casos de IRQ de dispositivo e IRQ de DMA
- > b) sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo
- no cubre el caso de E/S programada para dispositivos "sin estado" (ej: display 7 segmentos en las diapositivas)
- c) la consulta del estado del dispositivo por parte de la CPU se suele/puede incluir en E/S programada y en E/S por IRQ
- d) sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S

### 34. ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

- a) 3
- b) 4
- > c) 2
- d) 5

### 35. Se desea utilizar interrupciones vectorizadas para gestionar las interrupciones procedentes de 16 dispositivos. Si tenemos una CPU con un bus de datos de 8 bits, ¿es posible?

- a) Sí, empleando un decodificador
- > b) Sí, empleando un codificador
- c) Sí, empleando un multiplexor
- d) Ninguna de las respuestas anteriores es cierta

**36. Si el acceso directo a memoria se realiza mediante robo de ciclo:**

- a) es posible que la ejecución de una operación elemental de transferencia en el bus sea temporalmente detenida
- > b) es posible que la ejecución de una instrucción máquina sea temporalmente detenida
- c) es necesario que termine de ejecutarse la instrucción máquina actual para comenzar una transferencia por DMA
- d) ninguna de las anteriores es cierta

**37. Las técnicas principales de E/S son (señalar la respuesta falsa)**

- a) DMA (por acceso directo)
- > b) E/S cableada (hardwired)
- c) E/S programada
- d) IRQ (por interrupciones)

**38. ¿Qué método de control de acceso directo a memoria es preferible por velocidad (más rápida), economía (coste no prohibitivo) y conveniencia de diseño (compatible con memorias y sistemas actuales)?**

- a) DMA intercalado o transparente  
lento, para sistemas monoprocesador con sólo 1 maestro bus
- b) Robo de ciclo  
lento, para sistemas monoprocesador con sólo 1 maestro bus
- c) Memoria multipuerto
- > d) Transferencia de bloques o parada de CPU

**39. El fragmento de código:**

**poll: in a, 0x20**

**cmp a, \$0**

**jnz poll**

**load a, 0x11**

**out 0x21, a**

**corresponde a:**

- a) E/S por interrupciones
- > b) E/S programada con consulta de estado
- c) E/S por DMA
- d) E/S programada sin consulta de estado

**40. Algunas de las ventajas de la E/S aislada frente a la E/S mapeada en memoria son:**

- > a) Los programas pueden ser más rápidos, y las instrucciones de E/S son fácilmente reconocibles
- b) Facilita la protección de E/S, y el diseño de la CPU es menos complejo
- c) Las respuestas a y b son ciertas
- d) Las respuestas a y b son falsas

**41. Al método de interacción con los periféricos, en los que el procesador vigila periódicamente el estado de los dispositivos mediante una encuesta activa se le denomina:**

- a) interrupción
- > b) polling
- c) daisy-chain
- d) DMA

**42. Las interrupciones iniciadas por un dispositivo de E/S son normalmente:**

- a) espurias
- > b) externas
- c) software
- d) internas

**43. Una instrucción típica de entrada / salida tiene**

- a) tiene un argumento: un registro del procesador  
incluso aunque la transparencia 22 sugiera que es "IN puerto" y "OUT puerto", el argumento no es un registro
- b) no tiene ningún argumento
- c) tiene tres argumentos: un registro del procesador, una dirección de puerto de E/S y una dirección de memoria
- > d) tiene dos argumentos: un registro del procesador y una dirección de puerto de E/S  
claramente inspirado en el repertorio IA32 - no encuentro dónde aparece en las transparencias - "Ida a Data" en transparencia 40 es mapeado a memoria - evitar volver a poner esta pregunta en exámenes

**44. En la E/S controlada por interrupciones:**

- a) La CPU lee y comprueba el estado de los dispositivos de E/S (en el caso de consulta de estado).

poll? obligatorio con IRQ?

- b) El controlador de DMA envía una petición de interrupción a la CPU.

DMA?

- > c) La CPU transfiere el control a una rutina de servicio cuando recibe una interrupción.

Exacto, la ISR

- d) El controlador de DMA transfiere bloques de datos por el bus del sistema.

**45. Tipos de interrupción que suelen contemplar las CPUs comerciales actuales (señalar la opción incorrecta)**

- a) externas (IRQs hardware): generadas por un dispositivo externo a la CPU, activan la línea INTR# (o equivalente)

- b) software: generadas al ejecutar la instrucción INT (o equivalente)

- c) internas (excepciones o traps): generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, codop inválido, etc)

- > d) firmware (faults): generadas por el microcódigo de la CPU (segmentation fault, page fault, etc)

**46. Suponiendo que varios dispositivos comparten una única línea de solicitud de interrupción y que varios de ellos solicitan una interrupción al mismo tiempo, ¿qué dispositivo tendría mayor prioridad a la hora de ser atendidas sus peticiones?**

- a) Si se emplea una técnica de sondeo ("polling"), el dispositivo cuyo estado se consulte primero

- b) Si se emplea una técnica de encadenamiento ("daisy-chain"), el dispositivo al que primero se conecta la línea INTA# del procesador

- > c) Las respuestas a y b son ciertas

- d) Las respuestas a y b son falsas

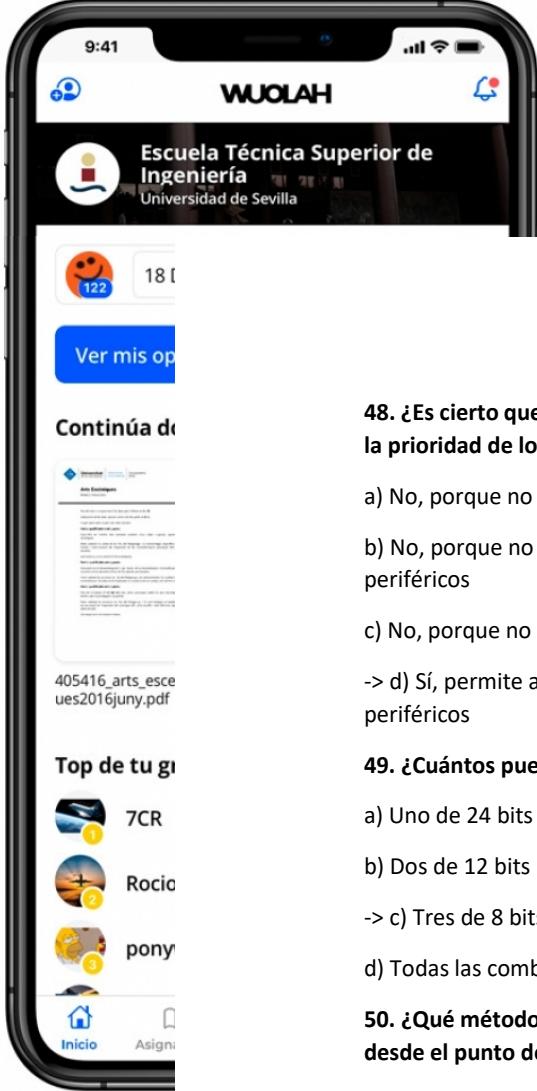
**47. ¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC (modo real) es cierta?**

- a) La tabla de interrupciones tiene un tamaño de 256 bytes

- b) Existen 1024 vectores de interrupción

- c) Cada vector de interrupción es una palabra de 16 bits

- > d) Todas las interrupciones se pueden generar por software



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**48. ¿Es cierto que la consulta de estado permite averiguar el estado de un periférico y alterar la prioridad de los periféricos?**

- a) No, porque no permite averiguar el estado de un periférico
  - b) No, porque no permite ni averiguar el estado de un periférico ni alterar la prioridad de los periféricos
  - c) No, porque no permite alterar la prioridad de los periféricos
- > d) Sí, permite averiguar el estado de un periférico y también alterar la prioridad de los periféricos

**49. ¿Cuántos puertos puede gestionar la interfaz de periféricos programable 8255?**

- a) Uno de 24 bits
- b) Dos de 12 bits
- > c) Tres de 8 bits
- d) Todas las combinaciones anteriores son válidas

**50. ¿Qué método de identificación de la fuente de una interrupción suele ser más económico desde el punto de vista hardware?**

- > a) La identificación mediante la técnica de sondeo
- b) La identificación mediante interrupciones vectorizadas
- c) El coste de las técnicas a) y b) es el mismo
- d) No es posible comparar el coste de los métodos a) y b)

**51. Respecto al sistema de Entrada / Salida, ¿cuál de las siguientes afirmaciones es incorrecta?**

- a) La CPU se comunica con el periférico por medio del controlador y de software de E/S.
- b) Un protocolo sirve para "ponerse de acuerdo" en cosas como velocidad, paridad, nº de bits, etc.
- c) Un controlador se encarga de la comunicación con la CPU.
- > d) La mayoría de los periféricos trabajan a velocidad muy superior a la CPU; por eso es necesario sincronizar.

**52. ¿Cuántos bits hacen falta como mínimo para implementar tres niveles de inhibición de interrupciones (general, nivel y máscara) en un sistema con cuatro niveles de interrupción?**

- a) 5
- b) 6
- c) 4
- > d) 7

**53. ¿Qué tipo de sincronización es más conveniente en el caso de tener dispositivos con distintos requisitos de temporización?**

- a) Síncrona
- > b) Asíncrona
- c) No se pueden conectar dispositivos con distintos requisitos de temporización
- d) Ninguna de las anteriores

**54. La conexión entre un dispositivo de E/S y el procesador mediante bus:**

- a) Requiere multiplexores y demultiplexores para las señales de datos
- > b) Permite conectar en paralelo varios dispositivos
- c) Requiere mucha circuitería
- d) Es difícil de expandir

**55. ¿Cuál de las siguientes afirmaciones es cierta?**

- a) La E/S en memoria emplea la patilla IO/M#
- > b) En E/S independiente, las instrucciones de acceso a memoria suelen ser más largas que las de E/S
- c) La E/S en memoria facilita la protección
- d) Ninguna de las anteriores es cierta

**56. ¿Cuál de las siguientes afirmaciones es incorrecta?**

- a) Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
- b) Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó)
- > c) Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S
- d) La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa polling para determinar el origen de la IRQ)

**57. La técnica de sondeo, escrutinio o "polling"...**

- a) Es incompatible con el daisy-chain
- b) No permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos
- c) En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción
- > d) Se utiliza para identificar la fuente de una interrupción

**58. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de  $2^{13}$  palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?**

- > a)  $2^{10}$
- b)  $2^{12}$
- c)  $2^{11}$
- d)  $2^{13}$

**59. ¿Cuántos controladores de interrupciones 8259 hacen falta como mínimo para manejar 17 líneas de interrupción?**

- a) 2
- b) 4
- > c) 3
- d) 5

**60. ¿Cuál de los siguientes elementos no forma parte del canal de un controlador de acceso directo a memoria?**

- a) Registro de dirección.
- > b) Registro de prioridades.
- c) Registro contador.
- d) Todos los elementos anteriores forman parte de un canal de DMA.

**61. Respecto a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida**

- a) todas ellas impiden que haya cortocircuito en el bus de datos  
selección lineal no lo garantiza - confía en diseño sensato
- > b) la selección lineal permitiría escribir un mismo dato a varios puertos E/S
- c) todas ellas impiden que haya cortocircuito en el bus de direcciones  
El arbitraje se encarga de eso
- d) usando decodificación centralizada es más fácil realizar expansiones al sistema de E/S

**62. Alguna de las siguientes NO es una técnica de E/S de las estudiadas en clase:**

- a) E/S mediante Acceso Directo a Memoria
- > b) E/S asíncrona
- c) E/S controlada por interrupciones
- d) E/S programada

**63. Un controlador de DMA suele ser programado con la siguiente información relativa a una operación de E/S:**

- a) tipo de operación, tamaño de bloque a transferir, dirección final de memoria
- > b) tipo de operación, tamaño de bloque a transferir, dirección inicial de memoria
- c) tipo de operación, dirección inicial de memoria, dirección final de memoria
- d) Ninguna de las anteriores respuestas es cierta

**64. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción no es cierta?**

- a) Es una bifurcación normalmente externa al programa en ejecución
- b) Su objetivo es reclamar la atención del procesador
- c) Sigue que se ejecute un programa específico para tratarla
- > d) Ninguna de las anteriores

**65. ¿Cuántos controladores de interrupciones 8259 hacen falta como mínimo para manejar 25 líneas de interrupción?**

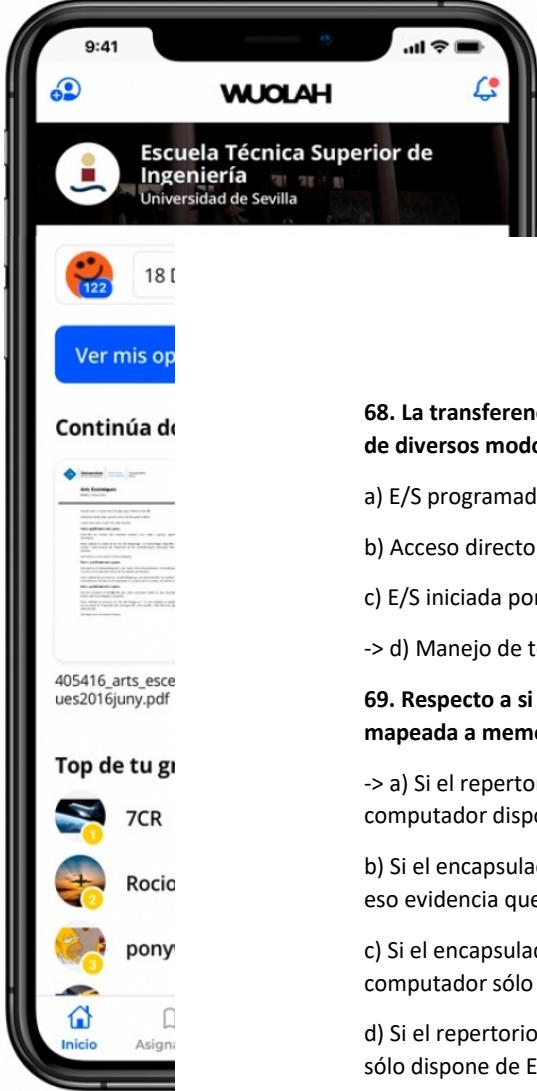
- a) 1
- b) 3
- c) 2
- > d) 4

**66. ¿Qué modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?**

- a) 3
- b) 1
- c) 0
- > d) 2

**67. La técnica de sondeo, escrutinio o "polling"...**

- a) Se utiliza para identificar el destino de una interrupción
- > b) Permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos
- c) En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción
- d) Todas las respuestas anteriores son falsas



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**68. La transferencia de datos en un computador y los dispositivos de E/S puede manejarse de diversos modos. Uno de los siguientes es falso; indíquelo:**

- a) E/S programada
- b) Acceso directo a memoria (DMA)
- c) E/S iniciada por interrupción
- > d) Manejo de todas las líneas del bus de control, paralizando la CPU

**69. Respecto a si un computador dispone de E/S independiente (separada) o usa E/S mapeada a memoria:**

- > a) Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada
- b) Si el encapsulado (chip) del procesador tiene patilla (pin) IO/M# (o patillas equivalentes), eso evidencia que el computador usa E/S mapeada a memoria
- c) Si el encapsulado del procesador no dispone de patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada
- d) Si el repertorio del procesador tiene instrucciones del tipo LOAD y STORE, el computador sólo dispone de E/S mapeada a memoria

**70. Con una línea de interrupción organizada en colector abierto:**

- a) se pueden conectar varios dispositivos de manera sencilla sin necesidad de circuitos combinacionales
- b) a la patilla del circuito integrado que genera esa línea hay que conectarle un transistor
- c) un dispositivo conectado a ella puede solicitar la interrupción poniendo a nivel bajo la línea
- > d) a) y c) son ciertas

**71. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de  $2^{13}$  palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?**

- a)  $2^{12}$
- b)  $2^{13}$
- c)  $2^{11}$
- > d)  $2^{10}$

**72. Alguna/s de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente es/son:**

- a) Las instrucciones de E/S son fácilmente reconocibles.
- b) Facilita la protección de E/S.
- > c) El diseño de la CPU es más sencillo.
- d) Todas las respuestas son ciertas.

**73. Supongamos dos CPU con idéntica anchura tanto en el bus de direcciones como en el de datos. Si una de ellas emplea E/S independiente y la otra mapeada en memoria, ¿cuál podrá acceder a una mayor cantidad de memoria?**

- > a) La CPU con E/S independiente
- b) La CPU con E/S mapeada en memoria
- c) Ambas podrán acceder a la misma cantidad de memoria
- d) Depende de la técnica de E/S utilizada

**74. Respecto a la E/S programada...**

- a) No todos los pasos requieren la ejecución de instrucciones por parte de la CPU
- b) La transferencia la realiza un procesador externo a la CPU
- c) Las dos primeras afirmaciones son ciertas
- > d) Las dos primeras afirmaciones son falsas

**75. ¿En qué tipos de técnicas de E/S la transferencia de información está bajo el control directo de la CPU?**

- > a) E/S programada y E/S controlada por interrupciones
- b) E/S programada y acceso directo a memoria
- c) E/S controlada por interrupciones y acceso directo a memoria
- d) Ninguna de las anteriores respuestas es correcta

**76. Respecto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR)**

- a) no es necesario salvar ninguno más, si el contador de programa y los flags de estado ya los salva la propia CPU como parte del mecanismo de interrupción
- b) se deben guardar los registros salva-invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA32), los registros salva-invocante ya los guarda el programa interrumpido
- > c) se deben guardar los registros que se modifiquen en la propia ISR. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar
- d) se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción

**77. Si varios dispositivos comparten una única línea de solicitud de interrupción:**

- a) se podría usar "polling" para identificar el origen de una interrupción
- b) se podría utilizar "daisy-chain" para establecer la prioridad de los dispositivos
- c) se podría utilizar interrupciones vectorizadas
- > d) todas son ciertas

**78. Alguna de las siguientes NO es una ventaja de la E/S independiente (separada, aislada)**

- > a) Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
- b) Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- c) Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- d) Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)

**79. Supongamos dos CPU con bus de direcciones de ancho idéntico. Si una de ellas emplea E/S independiente y la otra mapeada en memoria, ¿cuál podrá acceder a una mayor cantidad de memoria?**

- > a) La CPU con E/S independiente.
- b) Ambas podrán acceder a la misma cantidad de memoria.
- c) La CPU con E/S mapeada en memoria.
- d) Depende del bus de datos.

**80. ¿Con cuál de los siguientes dispositivos tendría sentido utilizar E/S programada sin consulta de estado?**

- > a) Salida a un display de 7 segmentos
- b) Entrada desde un disco duro
- c) Salida a una impresora
- d) Con ningún dispositivo tiene sentido

**81. Cuando se produce una interrupción hardware...**

- > a) Se guarda el estado y se ejecuta la rutina de interrupción asociada
- b) Se aborta la ejecución del programa actual generando un fallo de segmentación
- c) Se salta a la dirección de memoria indicada en la instrucción actual
- d) Se salta al principio del programa actual

**82. ¿A qué tipo de interrupciones pertenecen las condiciones de tiempo real y los fallos hardware?**

- a) Enmascarables
- > b) No enmascarables
- c) Puede ser tanto enmascarables como no enmascarables
- d) Ninguna de las respuestas anteriores es correcta

**83. Para determinar la causa de una interrupción se pueden usar las siguientes técnicas:  
(señalar la respuesta falsa)**

- a) consulta de estado, o polling
  - b) múltiples líneas de interrupción INT1#, INT2#...
  - c) interrupciones vectorizadas
- > d) línea de reconocimiento INTA#

**84. Utilizando E/S programada y como modo de direccionamiento selección lineal, ¿cuántos periféricos podrían conectarse a un 8086? Recordar que el 8086 disponía de E/S separada, con bus de direcciones de 20 bits para memoria y de 16 bits para E/S.**

- a)  $2^{16}$  periféricos
- > b) 16 periféricos
- c)  $2^{20}$  periféricos
- d) 20 periféricos

**85. Ventajas de la E/S independiente (separada, aislada) (señalar la opción incorrecta)**

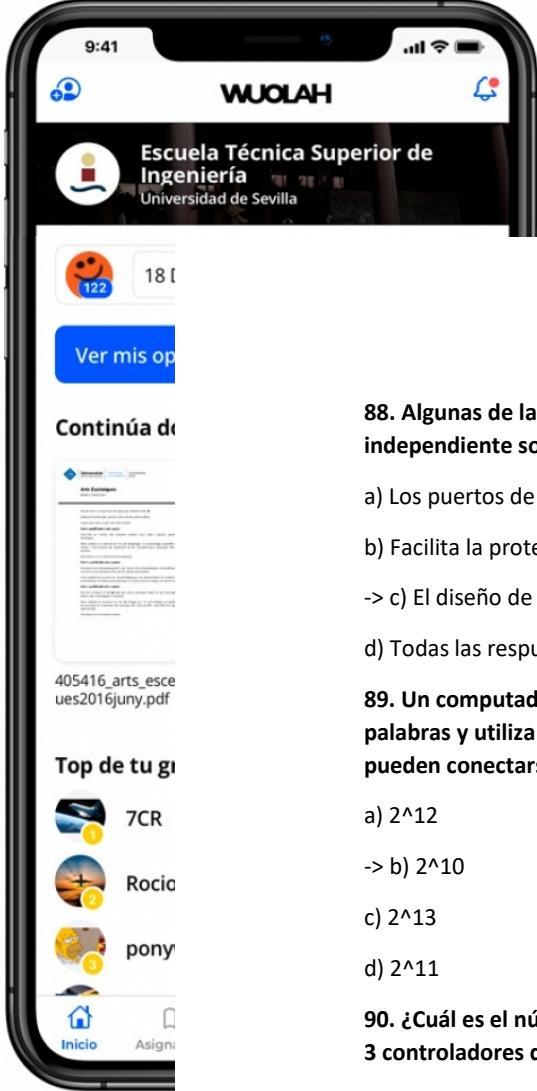
- a) mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- > b) diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
- c) decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
- d) protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)

**86. ¿Cuál de los siguientes es un registro de un controlador de DMA?**

- a) PC (Program Counter)
- > b) WC (Word Count)
- c) IR (Instruction Register)
- d) SP (Stack Pointer)

**87. Un procesador de 8 bits, ¿a cuántos puertos de E/S podrá acceder?**

- a) 8
- > b) Depende del método de selección de periféricos que emplee
- c) 256
- d) 65536



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**88. Algunas de las ventajas de la E/S mapeada en memoria frente a la E/S aislada o independiente son:**

- a) Los puertos de E/S no ocupan direcciones de memoria.
- b) Facilita la protección de E/S.
- > c) El diseño de la CPU es más sencillo.
- d) Todas las respuestas anteriores son ciertas

**89. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de  $2^{13}$  palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?**

- a)  $2^{12}$
- > b)  $2^{10}$
- c)  $2^{13}$
- d)  $2^{11}$

**90. ¿Cuál es el número máximo de niveles de interrupción que se pueden manejar utilizando 3 controladores de interrupciones programables 8259?**

- a) 24
- > b) 22
- c) 20
- d) 18

**91. ¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?**

- a) Registro de dirección
- b) Registro de órdenes
- c) Registro contador
- > d) Registro de prioridades

**92. ¿Cuántos puertos de E/S permite manejar la interfaz de periféricos programable 8255?**

- a) 2 puertos de 16 bits
- > b) 3 puertos de 8 bits
- c) 4 puertos de 32 bits
- d) Todas las respuestas anteriores son falsas

**93. ¿Cuál de las siguientes afirmaciones es cierta?**

- > a) La E/S independiente facilita la protección
- b) En E/S independiente, las instrucciones de acceso a memoria se emplean tanto para memoria como para E/S
- c) La E/S en memoria emplea la patilla IO/M#
- d) La E/S en memoria es mucho más rápida que la E/S independiente

**94. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es cierta?**

- a) Solicita que el procesador se aisle de los buses
- b) Su objetivo es incrementar el ancho de banda con el dispositivo
- > c) Es una bifurcación normalmente externa al programa en ejecución
- d) Permite realizar transferencias sin el control de un programa

**95. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es falsa?**

- a) Solicita que se ejecute un programa específico para tratarla
- > b) Ninguna de las afirmaciones es falsa
- c) Su objetivo es reclamar la atención del procesador
- d) Es una bifurcación normalmente externa al programa en ejecución

**96. ¿Qué instrucciones son típicas del repertorio de un procesador que sólo disponga de E/S mapeada en memoria?**

- a) IN, LOAD, OUT
- b) IN, LOAD, MOV
- > c) LOAD, MOV, STORE
- d) Ninguno de los anteriores

**97. De los siguientes grupos de técnicas de E/S, ¿cuáles están controlados por programa?**

- > a) E/S programada y E/S mediante interrupciones.
- b) E/S programada y E/S mediante acceso directo a memoria.
- c) E/S mediante interrupciones y E/S mediante acceso directo a memoria.
- d) Todas las respuestas anteriores son ciertas.

**98. Sobre las técnicas de transferencia en operaciones de E/S:**

- a) Pueden ser controladas por programa o por hardware
- b) Si se emplea E/S programada puede hacerse con consulta de estado o sin consulta de estado
- c) En el caso de utilizar E/S mediante DMA hace falta emplear un controlador de DMA
- > d) Todas las respuestas anteriores son ciertas

**99. En la práctica de E/S en Arduino, la instrucción SBI del repertorio del microcontrolador realiza...**

- a. una suma de un valor inmediato
- b. una resta de un valor inmediato
- > c. una operación de bits //P5 guión, pág.9-10
- d. un salto incondicional.

**100. En la práctica de E/S en Arduino, el DDRB es el...**

- a. segundo canal de memoria DDR
- > b. registro de dirección de datos del puerto B // P5 guión, pág.9-10
- c. registro de datos y direcciones B
- d. registro buffer de datos D

**101. En la práctica de E/S en Arduino, la instrucción SBIW del repertorio del microcontrolador realiza...**

- a. una suma de un valor inmediato
- > b. una resta de un valor inmediato //P5 guión, pág.9-12
- c. una operación de bits
- d. un salto incondicional

**102. En la práctica de E/S en Arduino, la instrucción CBI del repertorio del microcontrolador realiza...**

- a. una comparación
- b. un complemento a uno
- > c. una operación de bits //P5 guión, pág.9-10
- d. una llamada a subrutina

# TEST TEMA 6

**1. Una jerarquía de memoria consta de una cache de con una tasa de aciertos del 92% y 4 ns de tiempo de acceso y una memoria principal con una tasa de aciertos del 100% y 100 ns de tiempo de acceso. ¿Cuál es el tiempo promedio estimado de acceso a memoria?**

- a) 8 ns
- b) 6 ns
- c) 10 ns
- > d) 12 ns

**2. El ancho de banda de memoria es:**

- a) el número de bits que se pueden transferir entre ésta y la CPU en paralelo en una sola operación de lectura o escritura
- b) el tiempo que se tarda en transferir una palabra entre memoria y CPU
- c) el intervalo de frecuencias de reloj permitidas entre memoria y CPU
- > d) ninguna de las anteriores es cierta

**3. En un sistema con memoria de bytes y líneas de cache de 32 bytes, ¿dónde empieza el bloque de memoria que contiene la posición 0xAC72?**

- > a) 0xAC60
- b) 0xAC6E
- c) 0xAC70
- d) 0xAC72

**4. En un sistema con direcciones de 32bits, memoria de bytes, cache de 1MB asociativa por conjuntos de 4 vías y líneas de 64B, el campo etiqueta en el formato de dirección cache es de**

- a) 16 bits
- > b) 14 bits
- c) 10 bits
- d) 12 bits

**5. Una memoria que está organizada en palabras de 8 bits tiene una capacidad de 32 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?**

- > a) 12
- b) 4
- c) 8
- d) 32

**6. Considere un sistema de memoria para un procesador de 32 bits con caches separadas para código y datos. Suponga que el procesador direcciona la memoria por bytes y realiza accesos a palabras de 32 bits y que el espacio de direcciones es de 232 bytes. La cache de datos tiene las siguientes características: 64 KB de capacidad, asociativa por conjuntos con 2 vías, y bloques de 2 palabras. ¿Cuántos bits tiene el campo etiqueta de una dirección de memoria?**

- a) 13
- b) 11
- c) 15
- > d) 17

**7. ¿Cuál de las siguientes afirmaciones acerca de las memorias SRAM es falsa?**

- > a) Las operaciones de lectura son destructivas
- b) Son más veloces que las memorias RAM dinámicas
- c) El número de transistores necesario para implementar cada celda es mayor que en las memorias RAM dinámicas
- d) Los datos almacenados se mantienen por un tiempo indefinido mientras se mantenga la alimentación

**8. Un circuito SRAM con una capacidad de 256 Kbits tiene las patillas de direcciones A14 a A0. ¿Con cuál de las siguientes expresiones indicaría las características de capacidad en direcciones y datos del circuito?**

- > a) 32 K x 8
- b) 8 K x 32
- c) 64 K x 4
- d) 256 K x 1

**9. Un procesador accede en el instante de tiempo t a una posición de memoria  $d(t)$ . Poco tiempo después (en el instante de tiempo  $t+k$ ) accede a la posición anterior  $d(t)-1$ . Esos dos accesos son un ejemplo de...**

- > a) Localidad espacial
- $d(t+k)=d(t)-1$
- b) Localidad temporal
- c) No tiene nombre, ese tipo de localidad con incremento negativo ( $d(t)-1$ ) no se ha estudiado en clase
- d) No es una localidad, esa condición no guarda relación con el concepto de localidad

**10. Si el tiempo de acceso a la memoria cache es de 2 ns y el tiempo necesario para tratar un fallo de cache es de 80 ns, ¿cuál es la tasa de aciertos necesaria para que el tiempo medio de acceso al sistema de memoria sea de 10 ns?**

- a) 0.8
- b) 0.75
- > c) 0.9
- d) 0.95

**11. La política de correspondencia de una memoria cache con la mitad de conjuntos que líneas es:**

- a) Asociativa por conjuntos con 2 líneas  
con cuántas líneas? será cuántas vías!
- b) Directa con 2 líneas  
en directa no hay que decir vías ni líneas ni nada, basta saber los tamaños de línea y cache
- c) Totalmente asociativa de media vía  
media vía?
- > d) Asociativa por conjuntos de 2 vías

"vía" se refiere a cuántas "alternativas" (líneas) hay en un conjunto.

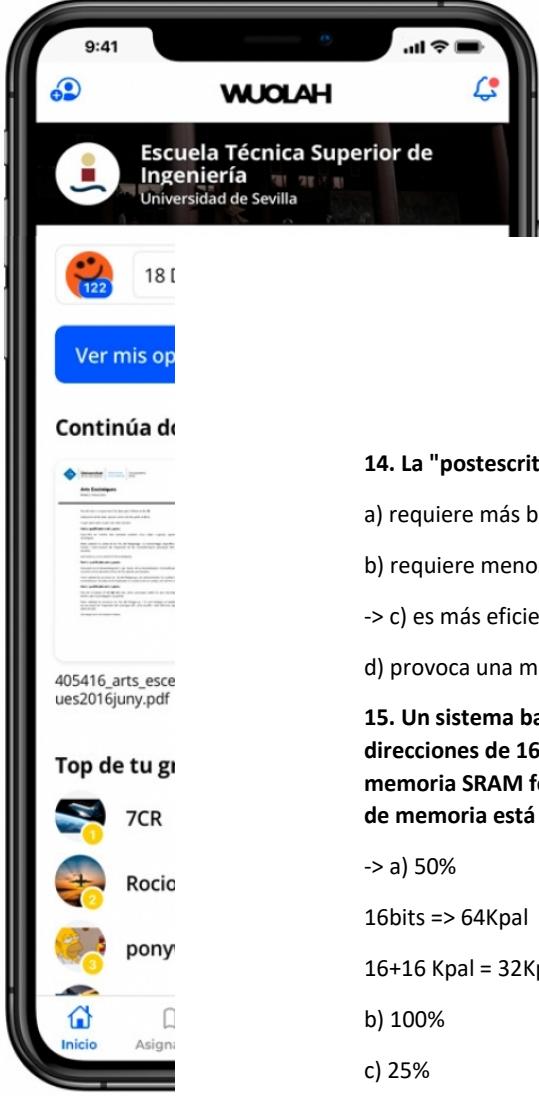
Además, todas las demás opciones tienen defectos

**12. ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM dinámicas es cierta?**

- a) Los datos permanecen en cada celda indefinidamente
- b) Las operaciones de lectura no son destructivas
- c) Las celdas de almacenamiento son complejas
- > d) Las operaciones de escritura sirven como operaciones de refresco

**13. ¿En qué se diferencian las estrategias de mantenimiento de coherencia en memoria "escritura directa" y "post-escritura"?**

- > a) En cuándo tiene lugar la actualización
- b) En cómo tiene lugar la actualización
- c) Tanto en a) como en b)
- d) Ni en a) ni en b)



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

#### 14. La "postescritura (write-back) marcada"

- a) requiere más bits de modificación ("bits sucios") cuando aumenta el número de vías
- b) requiere menos hardware que la "postescritura siempre"
- > c) es más eficiente que la "postescritura siempre"
- d) provoca una menor tasa de faltas que la "postescritura siempre"

#### 15. Un sistema basado en un microprocesador con un bus de datos de n bits y un bus de direcciones de 16 bits direcciona la memoria por palabras de n bits y dispone de una memoria SRAM formada por dos módulos de 16 K x n cada uno. ¿Qué porcentaje del mapa de memoria está ocupado por la SRAM?

- > a) 50%
- 16bits => 64Kpal
- $16+16 \text{ Kpal} = 32 \text{ Kpal} = 50\% \text{ } 64 \text{ Kpal}$
- b) 100%
- c) 25%
- d) 12,5%

#### 16. ¿Cuál de los siguientes es el ejemplo más acertado de localidad espacial?

- a) Reservar dinámicamente (malloc) espacio para una estructura o union
- b) Referenciar continuamente la misma variable local
- c) Iterar repetidamente el cuerpo de un bucle
- > d) Referenciar elementos de un array sucesivamente

#### 17. ¿En qué tipo de memorias coincide el tiempo de acceso y el tiempo de ciclo?

- > a) SRAM
- b) DRAM
- c) Tanto en a) como en b)
- d) Ninguna de las anteriores

#### 18. ¿Qué conjunto de componentes permite construir una memoria 256Mx32? (sin que sobren componentes)

- a) 16 chips 64Mx4
- > b) 32 chips 64Mx4
- c) 16 chips 64Mx16
- d) Ninguna de las anteriores

**19. ¿Cuál es el tamaño de la marca de cache si el bus de direcciones es de 48 bits (256 TB de memoria principal) y hay 8MB de cache L3, con un tamaño de línea de 64 B y correspondencia asociativa por conjuntos con 16 vías?**

- a) 13 bits
- b) 6 bits
- > c) 29 bits
- d) 48 bits

**20. La cache con correspondencia directa se puede considerar como un caso límite de la asociativa por conjuntos, en donde...**

- > a) solo hay 1 línea por conjunto
- b) solo hay 1 palabra por bloque
- c) solo hay 1 conjunto por cache
- d) ninguna de las anteriores

**21. ¿Qué necesitamos para construir una memoria de 1K x 8 bits?**

- a) 64 memorias de 128 x 1 bits
- b) 8 memorias de 512 x 2 bits
- > c) 8 memorias de 256 x 4 bits y un decodificador de 2 a 4
- d) Ninguna de las anteriores respuestas es cierta

**22. ¿Qué política de colocación en cache necesita más comparadores, la correspondencia asociativa por conjuntos o la correspondencia por sectores?**

- a) Correspondencia por sectores
- b) Correspondencia asociativa por conjuntos
- c) Depende de si es mayor el número de conjuntos o el número de sectores
- > d) Depende de si es mayor el número de bloques por conjunto o el número de sectores

**23. La cache es gestionada por:**

- a) el programador
- b) el sistema operativo
- > c) unidades de "manejo" (gestión) hardware
- d) ninguna es cierta

**24. La memoria cache en un sistema computador es:**

- a) Más rápida que la memoria principal
  - b) De menor capacidad que la memoria principal
- > c) a) y b) son correctas
- d) Ninguna de las anteriores es correcta

**25. La memoria DRAM:**

- a) Se inventó en la década de los 90
  - b) Necesita 6 transistores por cada celda
  - c) Es menos densa que la memoria SRAM
- > d) Se denomina dinámica porque para mantener almacenado un dato hay que recargarlo cada cierto tiempo en un ciclo de refresco

**26. En un sistema con memoria de bytes y líneas de cache de 64 bytes, ¿dónde empieza el bloque de memoria que contiene la posición 0xBEE3DE72?**

- a) 0xBEE3DE6E
- b) 0xBEE3DE70
- > c) 0xBEE3DE40
- d) 0x0EE3DE72

**27. Una SRAM de 32Kx8bit (256Kbit) puede venir organizada en 512 filas, dedicando por tanto al decodificador de columnas...**

- a) 8 bits
- > b) 6 bits
- c) 7 bits
- d) 9 bits

**28. Un computador con 10 bits en el bus de direcciones puede direccionar como máximo:**

- > a) 1024 palabras
- b) 1000 palabras
- c) 65535 palabras
- d) 65536 palabras

**29. ¿Cuál es el ancho del bus de direcciones de un chip DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?**

- a) 30
- > b) 15
- c) 16
- d) 20

Redacción original: "para una memoria DRAM"

Tal vez convendría modificar aún más el enunciado y preguntar el "nº de bits de direccionamiento en un chip DRAM..."

**30. ¿Cuál de las siguientes afirmaciones acerca de la jerarquía de memoria es \*FALSA\*?**

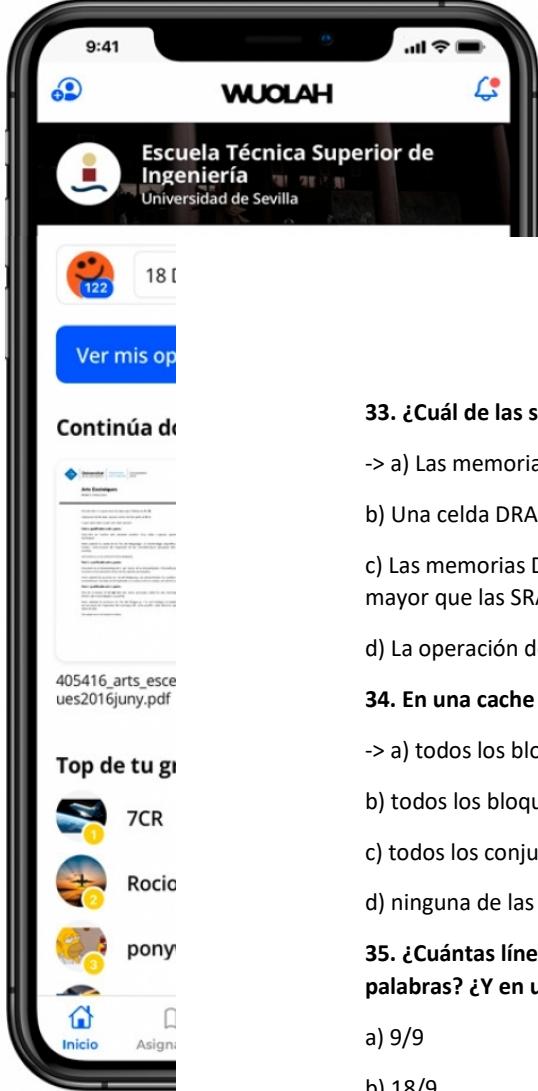
- a) Una memoria principal constituida por la tecnología más rápida es órdenes de magnitud más cara que la DRAM
- b) Acceder a los discos es órdenes de magnitud más lento que acceder a la RAM
- > c) La velocidad de acceso a la memoria principal ha crecido proporcionalmente a la velocidad del procesador
- d) Un computador puede tener una pequeña cantidad de memoria rápida además de una gran cantidad de memoria más lenta.

**31. ¿En qué tipo de refresco de memoria DRAM CAS# permanece a 0 después del ciclo de lectura o escritura precedente?**

- a) RAS# antes de CAS#
- b) Sólo RAS#
- > c) Refresco transparente
- d) Ninguna de las anteriores respuestas es correcta

**32. ¿Cuál de las siguientes afirmaciones sobre memorias cache es cierta? Recordar que llamamos "latencia" al tiempo transcurrido desde que se envía una dirección a la cache hasta que se obtiene el dato (suponiendo que se trata de un acierto), y depende por tanto del tiempo que emplee la circuitería hardware en obtenerlo. Suponer similares condiciones para ambas caches (tamaño de bloque, tamaño total, tecnología, frecuencia...)**

- a) Ambas tienen en general similar latencia y tasa de fallos
- > b) Las caches totalmente asociativas ofrecen tasas de fallo más bajas, mientras que las de correspondencia directa tienen mejor latencia
- c) Las caches de correspondencia directa tienen mejor latencia y tasa de fallos
- d) Las caches totalmente asociativas ofrecen mejor latencia, mientras que las de correspondencia directa tienen tasas de fallo más bajas



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

### 33. ¿Cuál de las siguientes afirmaciones es falsa?

- > a) Las memorias DRAM son en general mucho más rápidas que las SRAM
- b) Una celda DRAM sólo necesita un transistor y un condensador
- c) Las memorias DRAM presentan generalmente una capacidad de almacenamiento mucho mayor que las SRAM
- d) La operación de lectura de una celda DRAM es destructiva

### 34. En una cache asociativa por conjuntos, la vía i está constituida por:

- > a) todos los bloques i-ésimos de cada conjunto
- b) todos los bloques del conjunto i
- c) todos los conjuntos del bloque i
- d) ninguna de las anteriores es cierta

### 35. ¿Cuántas líneas de dirección son necesarias en un memoria RAM dinámica de 256 K palabras? ¿Y en una estática?

- a) 9/9
- b) 18/9
- > c) 9/18
- d) 18/18

### 36. Las técnicas write-through y write-back están relacionadas con

- > a) coherencia de cache
- b) arbitraje de buses
- c) métodos de E/S
- d) etapas de la unidad de control

### 37. ¿Cuál de las siguientes afirmaciones acerca de la memoria es \*FALSA\*?

- a) La memoria dinámica usa señales de control RAS# y CAS#
- b) Las celdas de memoria dinámica están constituidas por un transistor y un condensador
- > c) Las celdas de memoria estática tienen que ser constantemente refrescadas
- d) La memoria estática se emplea en las caches L1 y L2

**38. Se dispone de un computador cuyo tiempo medio de acceso al sistema de memoria cache y memoria principal es de 18 ns. Si la tasa de fallo de la cache es de 0,2 y el tiempo de acceso a la memoria principal es 50 ns. ¿Cuál es el tiempo de acceso a la cache?**

- a) 6 ns
- b) 24,4 ns
- c) 10 ns
- > d) 8 ns

**39. Un sistema de memoria tiene un tiempo medio de acceso de 10 ns por operación de lectura o escritura y un ancho de datos de 32 bits. ¿Cuál es el ancho de banda del sistema de memoria?**

- a) 2,5 ns
- b) 32 Mbytes por segundo
- > c) 400 millones de bytes por segundo
- d) 32 bits

**40. En un sistema Linux IA32, ¿cuántos enteros se podrían almacenar en una línea de cache, si la cache del procesador fuera de 4 KB, asociativa por conjuntos de 4-vías, y contuviera 4 conjuntos?**

- a) 16
- b) 128
- c) 32
- > d) 64

**41. ¿Cuál de los siguientes métodos para incrementar el ancho de banda de memoria es más económico?**

- a) Utilizar memorias de alta velocidad
- b) Duplicar el tamaño de la memoria
- > c) Organizar la memoria jerárquicamente
- d) Utilizar memorias asociativas

**42. Un módulo de memoria de 16 GB está formado por varios chips DRAM de 1024Mx4. ¿Cuántos chips DRAM necesita el módulo?**

- > a) 32
- $16\text{G} \times 8 = 16 \cdot (1024\text{M}) \times 2 \cdot (4)$
- b) 8
- c) 16
- d) 4

**43. Con 8 circuitos de memoria RAM de 1K x 8 se puede crear un memoria de:**

- a) 1K x 64
- b) 8K x 8
- c) 2K x 32

-> d) Todas las combinaciones anteriores son posibles

**44. En EC podemos usar la palabra directo para referirnos a... (señalar la opción incorrecta)**

-> a) dispositivo de almacenamiento secuencial directo (DASD)

DASD significa Direct Access Storage Device

- b) acceso directo a memoria
- c) cache con correspondencia directa
- d) modo de direccionamiento directo

**45. ¿Cuál de las siguientes es una idea fundamental de la jerarquía de memoria?**

a) Que dispositivos más grandes y lentos sirvan de cache para dispositivos más pequeños y rápidos

b) Crear una pequeña cantidad de almacenamiento que sea caro y lento

-> c) Que dispositivos más pequeños y rápidos sirvan de cache para dispositivos más grandes y lentos

d) Crear una gran cantidad de almacenamiento que sea caro y rápido

**46. En el siguiente código, ¿qué reordenamiento de los bucles muestra mejor localidad?**

```
// X, Y, Z ctes #define previo
```

```
int a[X][Y][Z]
```

```
int i, j, k, sum = 0;
```

```
for (i = 0; i < Y; i++)
```

```
 for (j = 0; j < Z; j++)
```

```
 for (k = 0; k < X; k++)
```

```
 sum += a[k][i][j];
```

a) El orden de los bucles no afecta a la localidad

b) j externo, k central, i interno

c) i externo, j central, k interno (el orden en que están ahora)

-> d) k externo, i central, j interno

**47. Una SRAM de 1Mx4bit (4Mbit) puede venir organizada en 2048 filas, dedicando por tanto al decodificador de columnas...**

-> a) 9 bits

$$1M = 2^{20} = 2^{11} \cdot 2^9 = 2048 \text{ filas} \cdot 512 \text{ columnas}$$

observar que no es 1Mbit organizado en 2048 filas (en cuyo caso la respuesta sería 7bits)

b) 7 bits

c) 8 bits

d) 6 bits

**48. Un computador emplea un sistema de memoria principal de 128 palabras y una memoria cache de 32 palabras. La organización de la memoria cache es totalmente asociativa y el tamaño de bloque es de 8 palabras. Se emplea el algoritmo de reemplazo FIFO. Si inicialmente la memoria cache está totalmente vacía, calcule el número de fallos cuando se lee la secuencia de direcciones de la memoria principal: 0000100, 1000001, 0000101, 0010011, 0100010, 1000100, 0000111.**

a) 3 fallos

b) 5 fallos

-> c) 4 fallos

d) 6 fallos

**49. ¿Cuál de las siguientes afirmaciones es cierta?**

a) La memoria SRAM es más lenta que la DRAM

b) La lectura en la memoria SRAM es destructiva

c) La memoria DRAM es más cara que la SRAM

-> d) Ninguna de las anteriores

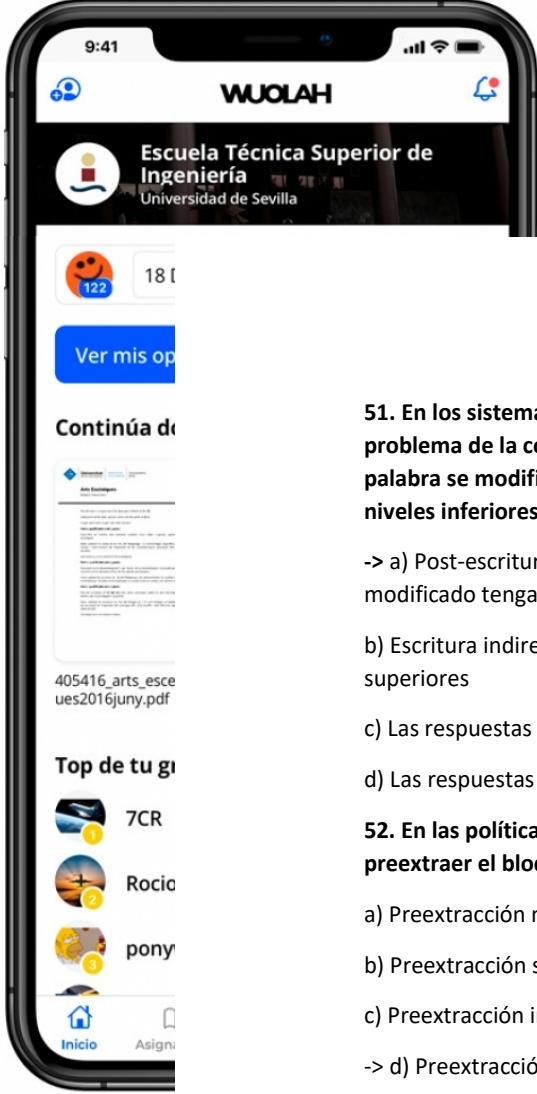
**50. En una memoria DRAM que permite el acceso en modo página se accede a la palabra 0x1234. Si emplea páginas de 256 palabras, ¿Cuál será la menor dirección a la que podremos acceder rápidamente?**

a) 0x1000

-> b) 0x1200

c) 0x1230

d) Otra



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**51. En los sistemas con una jerarquía de memoria dividida en varios niveles se da el problema de la consistencia o coherencia de los datos entre los distintos niveles. Si una palabra se modifica en un nivel, en algún momento habrá que traspasar ese cambio a los niveles inferiores (más lejanos al procesador). Para ello hay varias políticas:**

- > a) Post-escritura: se retrasa la actualización en los niveles inferiores hasta que el bloque modificado tenga que ser reemplazado
- b) Escritura indirecta: si se modifica una palabra, inmediatamente se modifican los niveles superiores
- c) Las respuestas a y b son ciertas
- d) Las respuestas a y b son falsas

**52. En las políticas anticipativas de extracción de cache, ¿cuál de ellas se caracteriza por preextraer el bloque  $i+1$  si se referencia al bloque  $i$  y se produce falta de bloque?**

- a) Preextracción marcada
- b) Preextracción siempre
- c) Preextracción indexada
- > d) Preextracción por falta

**53. Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta**

- a) 64 pastillas
- b) 8 pastillas
- > c) 16 pastillas
- x8 "a lo alto", x2 "a lo ancho"
- d) 32 pastillas

**54. Una cache de 256 B asociativa por conjuntos de 4-vías con líneas de 16 B tendría**

- a) 64 conjuntos
- b) ningún conjunto
- > c) 4 conjuntos
- $256\text{ B} = 2^8\text{ B} = 2^4\text{ líneas} \cdot 2^4\text{ B/lín}$
- $2^4\text{ lín} = 2^2\text{ conj.} \cdot 2^2\text{ vías}$
- d) 16 conjuntos

**55. El primer nivel de una jerarquía de memoria tiene una tasa de aciertos del 75% y las peticiones de memoria tardan 12 ns en completarse si dicha posición se encuentra en ese nivel y 100 ns si no es así. ¿Cuál es el tiempo medio de acceso de la jerarquía?**

- > a) 34 ns
- b) 25 ns
- c) 112 ns
- d) 88 ns

**56. ¿En qué tipo de ciclo de refresco se hace RAS# = 0?**

- a) Sólo RAS#
- b) CAS# antes de RAS#
- c) Refresco transparente
- > d) En todos los anteriores

**57. Sea un computador de 32 bits que dispone de una memoria cache de 512 KB y líneas de 64 bytes. ¿Cuántas líneas tiene la cache?**

- a) 1024
- b) 64
- c) 65536
- > d) 8192

**58. Para direccionar una memoria de 16K x 16 necesitamos un bus de direcciones de:**

- > a) 14 bits
- b) 16 bits
- c) 4 bits
- d) Otro valor

**59. Sea una cache asociativa por conjuntos de 4-vías. ¿Cuál de las siguientes afirmaciones es cierta?**

- > a) La cache tiene 4 líneas por conjunto
- b) La cache tiene 4 conjuntos por bloque
- c) La cache tiene 4 bloques por línea
- d) La cache tiene 4 conjuntos por línea

**60. Utilizar una cache en el mismo chip del procesador:**

- > a) reduce los tiempos de ejecución
- b) aumenta el tamaño de los bloques enviados entre cache y procesador
- c) aumenta la tasa de aciertos
- d) reduce el tamaño del bus

**61. ¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, los elementos cercanos a él serán referenciados pronto"?**

- > a) Localidad espacial
- b) Localidad secuencial
- c) Localidad temporal
- d) Ninguna de las respuestas anteriores es correcta

**62. ¿A qué tipo de memoria cache corresponde la siguiente afirmación: "permite que cualquier dirección se pueda almacenar en cualquier marco de bloque de cache"?**

- a) Con correspondencia directa
- > b) Totalmente asociativa
- c) Asociativa por conjuntos
- d) Ninguna de las anteriores

**63. Se desea construir una memoria de SRAM de tamaño 3G X 8 a partir de elementos de memoria SRAM más pequeños. Cuál de las siguientes soluciones sería correcta:**

- a) 256 chips de 16Mx 1 bits
- b) 16 chips de 512 M x 2 bits
- > c) 12 chips de 512M x 4 bits
- d) Ninguna de las anteriores es correcta

**64. En una jerarquía de memoria, a medida que nos alejamos del procesador:**

- a) el tiempo de transferencia disminuye
- > b) el tamaño de la unidad de transferencia entre dos niveles aumenta
- c) el tamaño de la memoria disminuye
- d) el coste por byte aumenta

**65. Un sistema tiene una cache asociativa por conjuntos de 2-vías con 16 conjuntos y líneas de 64B. ¿A qué conjunto le corresponde el byte con dirección 0xdeadbeef?**

- a) 13
- b) 14
- c) 7
- > d) 11

**66. Un programa crea en memoria una larga secuencia de números de forma consecutiva. ¿Qué tipo de estrategia de mantenimiento de coherencia es más eficiente para ejecutar este programa en un sistema con jerarquía de memoria?**

- a) Escritura directa ("write-through")
- > b) Post-escritura ("write-back")
- c) Tanto a) como b) son igual de eficientes
- d) No puede saberse qué técnica es mejor

**67. Variación de los parámetros de los distintos niveles en una jerarquía de memoria (señalar la opción incorrecta)**

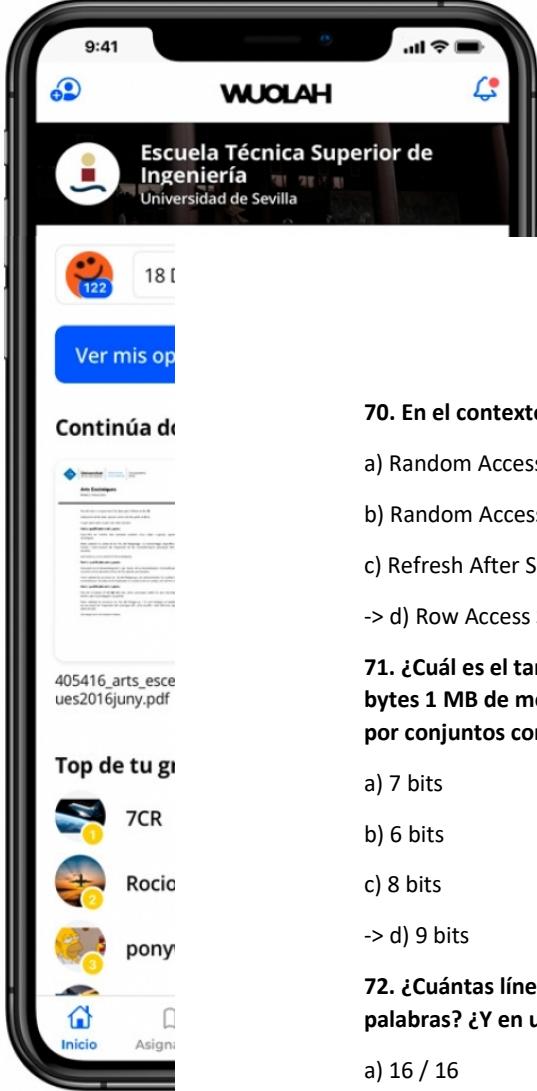
- a) ancho de banda:  $b_i \geq b_{i+1}$
- > b) unidad de transferencia:  $x_i \geq x_{i+1}$
- c) tiempo de acceso:  $t_i \leq t_{i+1}$
- d) tamaño del nivel:  $s_i \leq s_{i+1}$

**68. ¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, volverá a ser referenciado pronto"?**

- a) Localidad espacial
- b) Localidad lógica
- > c) Localidad temporal
- d) Ninguna de las respuestas anteriores es correcta

**69. ¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, volverá a ser referenciado pronto"?**

- a) Localidad secuencial
- b) Localidad espacial
- c) Localidad iterativa
- > d) Localidad temporal



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**70. En el contexto de las DRAM, RAS significa:**

- a) Random Access Strobe (muestreo de acceso aleatorio)
- b) Random Access Shot (disparo de acceso aleatorio)
- c) Refresh After Select (refresco después de selección de la memoria)
- > d) Row Access Strobe (muestreo de acceso a filas)

**71. ¿Cuál es el tamaño de la etiqueta de cache en un ordenador capaz de direccionar por bytes 1 MB de memoria principal y 32 KB de memoria cache y correspondencia asociativa por conjuntos con 32 bytes por línea y 16 líneas por conjunto?**

- a) 7 bits
- b) 6 bits
- c) 8 bits
- > d) 9 bits

**72. ¿Cuántas líneas de dirección son necesarias en un memoria RAM dinámica de 64 K palabras? ¿Y en una estática?**

- a) 16 / 16
- b) 16 / 8
- > c) 8 / 16
- d) 8 / 8

**73. ¿Cuáles de las siguientes direcciones de memoria podrían estar simultáneamente en una memoria cache de 256 palabras con 16 palabras por bloque y con correspondencia directa ?**

- a) 0x0000 y 0xFFFF
- > b) 0xABAB y 0xABAC
- c) 0x08E3 y 0x74E1
- d) Ninguna de las combinaciones anteriores

**74. Respecto al refresco de memorias DRAM, ¿cuál de las siguientes afirmaciones es falsa?**

- a) Se precisa una circuitería auxiliar, externa al chip DRAM o integrada en él, que produzca ciclos de refresco.
- b) Los ciclos de refresco deben producirse cada pocos ms (milisegundos).
- c) Los chips DRAM refrescan automáticamente la fila accedida en cualquier ciclo de lectura o escritura.
- > d) Una operación de refresco consiste en dar un impulso /CAS junto con una dirección de columna.

**75. En una cache con 64 bytes de longitud de línea, ¿qué bits de una dirección de memoria de 64 bits se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección? (Memoria direccionable por bytes)**

- a) [8...6]
- b) [11...6]
- c) [5...3]
- > d) [5...0]

**76. Si se necesitan 60 ns para escribir una palabra de datos de cache en memoria principal y cada bloque de cache tiene 8 palabras, ¿cuántas veces "seguidas" (sin que haya reemplazo) se tiene que escribir en un mismo bloque para que una cache de postescritura sea más eficiente que una de escritura inmediata?**

- a) Más de 8 veces
- > b) No se puede responder con los datos proporcionados
- c) La cache de postescritura no puede ser más eficiente que la de escritura inmediata
- d) La cache de postescritura siempre será más eficiente que la de escritura inmediata

**77. ¿Cuántas patillas de dirección tiene una memoria DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?**

- a) 30
- > b) 15
- c) 16
- d) 20

**78. Una memoria SRAM tiene una capacidad de 64 Kbits y utiliza 12 líneas para direccionamiento. Indique cuál es el tamaño de palabra de dicha memoria:**

- > a) 16 bits
- b) 8 bits
- c) 32 bits
- d) 64 bits

**79. Los módulos de memoria dinámica compactos que suelen usarse en los portátiles se denominan:**

- > a) SODIMM
- b) SLIM
- c) SIMM
- d) MIN

**80. ¿Cuál de las siguientes afirmaciones acerca de una jerarquía de memoria es cierta?**

- > a) Para aumentar la eficiencia se transfieren bloques completos
- b) Toda la información que la CPU necesita está en el nivel 1
- c) Si una palabra no se encuentra en el tercer nivel entonces se busca en el segundo nivel
- d) Todas las afirmaciones anteriores son falsas

**81. La tasa de aciertos  $A_i$  del nivel  $i$  de una jerarquía de memoria no depende de**

- a) La unidad de la transferencia de información  $x_i$  entre el nivel  $i$  y el  $i+1$ .
- b) La capacidad (tamaño)  $s_i$  del nivel  $i$ .
- > c) El ancho de banda  $b_i$  del nivel  $i$ .

Todas las demás se mencionan explícitamente en Tema 6 tr.24

- d) La estrategia de administración de memoria.

**81. ¿Cuál es el tamaño de la marca de cache en un microprocesador con 32 KB de memoria cache asociativa por conjuntos con 16 palabras de 32 bits por bloque y 8 bloques por conjunto, si el microprocesador es capaz de direccionar 1 MB de memoria principal (memoria de bytes)?**

- > a) 8 bits
- b) 7 bits
- c) 10 bits
- d) 6 bits

**82. ¿Qué dice la ley de Moore?**

- a) Que el tamaño de los transistores se duplica cada 18 meses.
- b) Que la memoria de los ordenadores se duplica cada 18 meses.
- > c) Que las prestaciones de los transistores en un chip se duplican cada 18 meses.
- d) Todas las respuestas son ciertas.

**83. La política de correspondencia de una memoria cache con 1 único conjunto es:**

- > a) Totalmente asociativa

todas las líneas son distintas vías del único conjunto -> cualquier bloque de memoria puede ir a cualquier marco de cache. Ver Tema 6 tr.137

- b) Asociativa por conjuntos con una única línea
- c) Directa
- d) Asociativa por conjuntos de una única vía

**84. ¿Cuántas líneas de dirección (patillas) son necesarias para direccionar un chip de memoria DRAM de 4096 x 4?**

- a) 11
- > b) 6
- c) 12
- d) 10

**85. ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM estáticas es falsa?**

- a) Los datos almacenados se mantienen por un tiempo indefinido
- b) Las operaciones de lectura no son destructivas
- c) Son más veloces que las memorias RAM dinámicas
- > d) El número de transistores necesario para implementar cada celda es menor que en las memorias RAM dinámicas

**86. Suponga el siguiente diseño cache/memoria: direcciones de 16 bits, direccionamiento por bytes, tamaño de cache 256 bytes, tamaño de bloque 8 bytes, tamaño de etiqueta 11 bits. ¿Cuál es la asociatividad (cuántas vías hay en la cache)?**

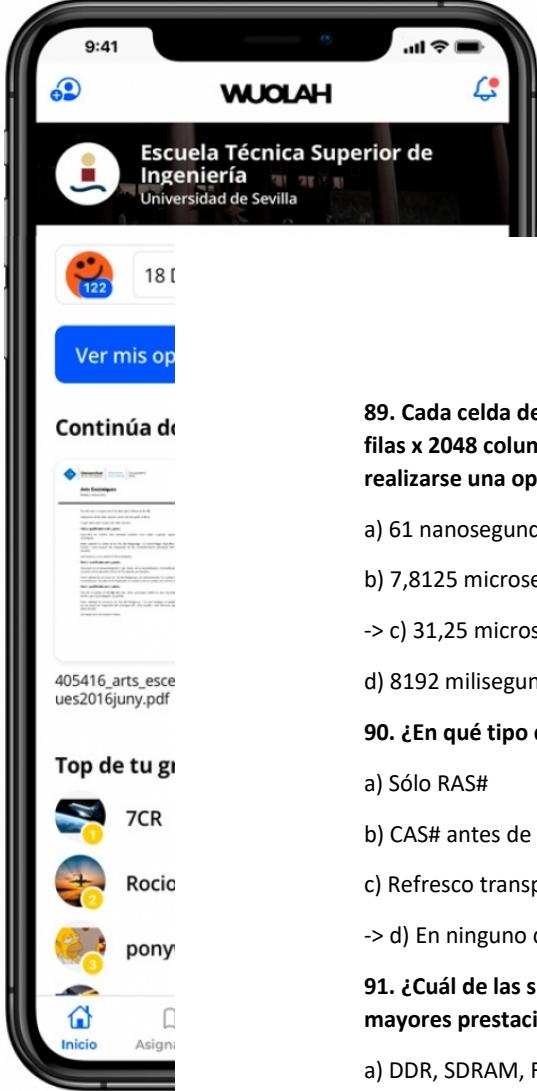
- a) 4
- > b) 8
- c) 2
- d) 5

**87. Sólo una de las siguientes afirmaciones sobre memorias ROM es correcta. ¿Cuál?**

- a) Una PROM (Programmable ROM) se puede grabar usando un dispositivo programador que selectivamente funde contactos aplicándoles altas temperaturas mediante diminutas cabezas soldadoras ("equipo de puntas")
- b) Una EEPROM (Erasable EPROM) se puede grabar (eléctricamente), y borrar (usando rayos ultravioleta)
- > c) Para fabricar una ROM se deben conocer los datos que se desea que almacene
- d) Una EPROM (Electrically Progr. ROM) se puede grabar eléctricamente, sin fundir contactos, pero no se puede borrar

**88. En un sistema con memoria de bytes, ¿cuál sería el tamaño de una línea de cache, si la cache del procesador fuera de 4MB, asociativa por conjuntos de 16-vías, y contuviera 4096 conjuntos?**

- > a)  $64\text{ B} // 4\text{MB} = 2^{22}\text{ B} = 2^{12}\text{ conj.} \cdot 2^4\text{ lin./conj. (vías)} \cdot 2^6\text{ B/lin.}$
- b) 16 B
- c) 128 B
- d) 32 B



# Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the  
App Store

GET IT ON  
Google Play

**89. Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?**

- a) 61 nanosegundos
- b) 7,8125 microsegundos
- > c) 31,25 microsegundos
- d) 8192 milisegundos

**90. ¿En qué tipo de ciclo de refresco RAS# permanece a 1?**

- a) Sólo RAS#
- b) CAS# antes de RAS#
- c) Refresco transparente
- > d) En ninguno de los anteriores

**91. ¿Cuál de las siguientes secuencias de tipos de memoria está ordenada de menores a mayores prestaciones?**

- a) DDR, SDRAM, FPM
- b) EDO, SRAM, FPM
- > c) FPM, EDO, RDRAM
- d) SDRAM, DDR, EDO

**92. En una memoria organizada en forma jerárquica, ¿qué suele ocurrir con respecto al tamaño de las unidades de transferencia entre niveles, conforme se baja de nivel hacia el procesador?**

- a) Aumenta
- > b) Disminuye
- c) Aumenta en algunos niveles y disminuye en otros
- d) Todas las posibilidades anteriores suelen darse

**93. En un sistema con memoria de bytes y líneas de cache de 64 bytes, ¿qué bits de una dirección de memoria se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección?**

- > a) Los 6 bits menos significativos
- b) Los 6 bits más significativos
- c) Los 4 bits menos significativos
- d) Los 4 bits más significativos

**94. Para diseñar una memoria con ancho de palabra  $k \cdot m$  (y mismo nº palabras que los módulos) a partir de módulos con ancho de palabra  $m$ , se utilizan  $k$  módulos  
(mediante \_\_\_\_\_ se representa subíndice)**

-> a) repartiendo líneas datos: el 1º se conecta a  $D_0...D_{m-1}$ , el 2º a  $D_m...D_{2m-1}$ , etc

Corregida errata  $D_{2m-1}$

b) repartiendo las líneas de datos entre los  $k$  módulos: el primero se conecta a  $D_0...D_{k-1}$ , el segundo a  $D_k...D_{2k-1}$ , etc

c) repartiendo líneas dirección: el 1º se conecta a  $A_0...A_{m-1}$ , el 2º a  $A_m...A_{2m-1}$ , etc

d) repartiendo las líneas de dirección: el 1º se conecta a  $A_0...A_{k-1}$ , el 2º a  $A_k...A_{2k-1}$ , etc

**95. Se dispone de un circuito integrado que actúa como módulo básico de memoria de  $8K \times 4$ . ¿Qué circuitos necesitamos para construir una memoria de  $16K \times 8$  para usar con un bus de direcciones de 14 bits?**

a) 2 módulos de memoria

b) 2 módulos de memoria y un decodificador

c) 4 módulos de memoria

-> d) 4 módulos de memoria y un inversor

**96. Una memoria que está organizada en palabras de 16 bits tiene una capacidad de 64 Kbits. ¿Cuántas palabras tiene?**

a) 65536

b) 4000

-> c) 4096

d) 64000

**97. Una puerta AND con 16 entradas conectada a un bus de direcciones de 16 bits, con todos los bits negados excepto A10 y A6, permite seleccionar un dispositivo (con CS activa en alta) en la dirección:**

-> a) 0x0440

b) 0x0220

c) 0xFDDF

d) 0xFBBF

**98. En la cache L1 de instrucciones, la tasa de fallos:**

-> a) Siempre tiende a disminuir si el tamaño total de L1 crece

b) Siempre tiende a crecer si el tamaño total de L1 crece

c) Siempre tiende a disminuir si el número de vías disminuye

d) Siempre tiende a crecer si el número de vías crece

**99. Las celdas de memoria estática...**

- a) son más pequeñas y lentas que las celdas de memoria dinámica
- b) almacenan la información en un condensador
- > c) mantienen las informaciones almacenadas por tiempo indefinido mientras se mantenga la alimentación
- d) Ninguna de las respuestas anteriores es cierta

## Preguntas típicas de los test

[T1.4]

1. ¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?
  - a) La arquitectura del procesador.
  - b) La frecuencia de reloj del procesador.
  - c) El número medio de ciclos de reloj por instrucción.
- d) El resultado de la ejecución de un conjunto de programas de prueba.

[T5.1]

2. ¿Qué instrucciones son típicas del repertorio de un procesador que sólo disponga de E/S mapeada en memoria?

- a) IN, LOAD, OUT
- b) IN, LOAD, MOV
- c) LOAD, MOV, STORE
- d) Ninguno de los anteriores

[T2.1.2]

3. ¿Cómo se devuelve en ensamblador x86 Linux gcc el valor de retorno de una función al terminar ésta?

- a) Se almacena en pila justo encima del (%ebp) del invocado
- b) La instrucción RET lo almacena en un registro especial de retorno
- c) Por convención se guarda en %eax
- d) Se almacena en pila justo encima de los argumentos de la función

[T3.3]

4. ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?

- a) Un multiplexor.
- b) Una memoria.
- c) Un contador.
- d) Un registro.

[T5.3]

5. ¿Cuántos bits hacen falta como mínimo para implementar tres niveles de inhibición de interrupciones (general, nivel y máscara) en un sistema con cuatro niveles de interrupción?

- a) 6
- b) 4
- c) 5
- d) 7

[T2.2.1]

6. Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?

- a) El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
- b) El factor de escala S puede ser 1, 2, 4, 8
- c) Los registros base e índice (Rb y Ri) pueden ser cualesquiera de los 8 registros enteros (EAX...ESP)
- d) El desplazamiento D también puede ser el nombre de una variable (que se traduce por su dirección, de 4bytes)

[T2.1.4]

7. ¿Cuál afirmación es FALSA en arquitecturas x86-64?

- a) El tamaño de un puntero es 64 bits
- b) El tamaño de los registros es 64 bits
- c) El tamaño de un double es 64 bits
- d) El tamaño de las posiciones de memoria es 64 bits

[T2.2.3]

8. ¿Cuál de las siguientes parejas de mnemotécnicos de ensamblador IA32 corresponden a la misma instrucción máquina?

- a) JZ (saltar si cero), JE (saltar si igual)
- b) CMP (comparar), SUB (restar)
- c) JC (saltar si acarreo), JL (saltar si menor, para números con signo)
- d) SAR (desplazamiento aritmético a la derecha) / SHR (desplazamiento lógico a la derecha)

[T5.1]

9. ¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?

- a) En las transferencias síncronas
- b) En las transferencias asíncronas
- c) En ambas
- d) En ninguna

[T1.1]

10. ¿Cuál de las siguientes direcciones está alineada a double (8-byte)?

(Al no poder escribir el 2 como subíndice, aclaramos que ")2" indica binario)

- a) 1110110101110111)2
- b) 1110110101110100)2
- c) 1110110101110000)2
- d) Ninguna de ellas

[T1.5]

11. ¿Cuál es la característica tecnológica principal de la segunda generación de computadores?

- a) Las válvulas
- b) Los circuitos integrados
- c) Los transistores
- d) La gran integración de los circuitos (VLSI)

[T2.4.3]

12. Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.

```
struct RECORD {
 long value2;
 int ref_count;
 char tag[4];
};
```

```

struct NODE {
 double value;
 struct RECORD record;
 char string[8];
};

```

También se declara una variable global "my\_node" como sigue:

```
struct NODE my_node;
```

Si la dirección de my\_node es 0x601040, ¿cuál es el valor de &my\_node.record.tag[1]?

- a) 0x601050
- b) 0x601054
- c) 0x601055
- d) Ninguna de las anteriores

[T2.1.4]

13. Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA32 como en x86-64. Sólo una de ellas es FALSA. ¿Cuál?

- a) El tamaño de un long es 4 bytes
- b) El tamaño de un int es 4 bytes
- c) El tamaño de un unsigned es 4 bytes
- d) El tamaño de un short es 2 bytes

[T5.1]

14. ¿Cuál de las siguientes funciones no corresponde a un controlador (interfaz) de E/S?

- a) Almacenamiento de programas
- b) Comunicación con el microprocesador
- c) Comunicación con el dispositivo
- d) Almacenamiento temporal de datos

[T6.2]

15. ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM dinámicas es cierta?

- a) Las celdas de almacenamiento son complejas
- b) Las operaciones de lectura no son destructivas
- c) Los datos permanecen en cada celda indefinidamente
- d) Las operaciones de escritura sirven como operaciones de refresco

[T5.2]

17. ¿Con cuál de los siguientes dispositivos tendría sentido utilizar E/S programada sin consulta de estado?

- a) Salida a un display de 7 segmentos
- b) Entrada desde un disco duro
- c) Salida a una impresora
- d) Con ningún dispositivo tiene sentido

# ENCENDER TU LLAMA CUESTA MUY POCO



[T5.3]

18. ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

- a) 2
- b) 4
- c) 5
- d) 3

[P2A2]

19. ¿Cuál de los siguientes grupos de instrucciones IA32 sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

- a) AND, OR, XOR
- b) CMP, TEST
- c) IMUL, IDIV
- d) ADC, SBB

[T2.2.4]

20. ¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

- a) JMP dir
- b) CALL dir
- c) RET
- d) JNE dir

[T6.1]

21. ¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, los elementos cercanos a él serán referenciados pronto"?

- a) Localidad espacial
- b) Localidad secuencial
- c) Localidad temporal
- d) Ninguna de las respuestas anteriores es correcta

[T6.2]

22. Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?

- a) 31,25 microsegundos
- b) 8192 milisegundos
- c) 7,8125 microsegundos
- d) 61 nanosegundos

[T6.5]

23. Si se necesitan 60 ns para escribir una palabra de datos de cache en memoria principal y cada bloque de cache tiene 8 palabras, ¿cuántas veces "seguidas" (sin que haya reemplazo) se tiene que escribir en un mismo bloque para que una cache de postescritura sea más eficiente que una de escritura inmediata?

- a) La cache de postescritura no puede ser más eficiente que la de escritura inmediata
- b) La cache de postescritura siempre será más eficiente que la de escritura inmediata
- c) No se puede responder con los datos proporcionados
- d) Más de 8 veces

BURN.COM

#StudyOnFire

**BURN**  
ENERGY DRINK

WUOLAH

[T2.4.1]

24. ¿Cómo se devuelve en ensamblador x86 64 Linux gcc el valor de retorno de una función long int al terminar ésta?

- a) La instrucción RET lo almacena en un registro especial de retorno.
- b) Por convención se guarda en %eax.
- c) Se almacena en pila justo encima de los argumentos de la función.
- d) Ninguna de esas formas es la correcta.

[T1.2]

25. En la ejecución de una instrucción...

- a) siempre se altera el registro de estado
- b) la UC activa las señales de control que envía por el bus de direcciones
- c) la ALU realiza las operaciones aritméticas y lógicas
- d) el registro de instrucción se va incrementando para apuntar a la siguiente instrucción

[T3.3]

26. Sea un formato de microinstrucción que incluye dos campos independientes de 10 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorran en cada microinstrucción?

- a) 10
- b) 13
- c) 14
- d) 9

[T3.3]

27. Un computador tiene una memoria de control de 16000 palabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorraremos usando nanoprogramación en lugar de microprogramación?

- a) 3744250
- b) 259206
- c) 287935
- d) ninguno de los resultados anteriores es exacto

[T2.1.4]

28. Estudiando el listado de una función C presuntamente compilada con gcc en modo 64bit (x86-64), nos dicen que la instrucción "movl (%rdi), %eax" carga en el registro EAX el valor adonde apunta el primer argumento.

- a) Está mal, porque el primer argumento de una función C no se pasa en RDI
- b) Está bien, y pone a cero los 32 bits más significativos de RAX
- c) Está mal, porque EAX no se puede usar en modo 64bit, debería ser RAX
- d) Está mal, porque EAX no se carga con ningún valor

[T2.3.1]

29. Respecto a la convención de llamada usada en Linux/gcc:

- a) Todos los registros pueden ser modificados libremente por todas las subrutinas  
Hay registros modificables, otros que deben ser restaurados, y las subrutinas
- b) anidadas deben respetar los registros modificables que están en uso por otras subrutinas
- c) Una subrutina que modifique algún registro debe restaurar su valor anterior antes de retornar

Hay registros que pueden ser modificados libremente por las subrutinas, y otros

- d) que, si se modifican, se deben restaurar posteriormente. Y también hay registros especiales

[T2.4.2]

30. Se ha declarado en un programa C la variable int val[5]={1,5,2,1,3} . ¿Cuál de las tres primeras opciones (a, b, o c) es FALSA?

- a)  $\text{val}[1] == 1$
- b)  $\&\text{val}[3] == \text{val} + 3$
- c)  $\text{sizeof}(\text{val}) == 20$

No se puede marcar ninguna de ellas, todas (a, b y c) son ciertas

- d) Antes ponía "Todas son ciertas". Esperamos que con la nueva redacción sea más fácil identificarla como la opción de descarte en un test con respuesta única.

Cambio de redacción. Antes ponía "cuál de las siguientes" y algunos estudiantes indicaban que d) era problemática (al ser falsa también), en lugar de identificarla como la opción de descarte en un test con respuesta única. Esperamos que la nueva redacción deje de dar a los estudiantes la sensación de que se les pretende confundir con sutilezas.

[T6.5]

31. ¿A qué tipo de memoria cache corresponde la siguiente afirmación: "permite que cualquier dirección se pueda almacenar en cualquier marco de bloque de cache"?

- a) Con correspondencia directa
- b) Totalmente asociativa
- c) Asociativa por conjuntos
- d) Ninguna de las anteriores

[T3.3]

34. Una posible codificación en microinstrucciones de la instrucción CALL X es:

- a)  $\text{PC} = X ; \text{SP} = \text{SP}-1 ; m[\text{SP}] = \text{PC}$
- b)  $\text{SP} = \text{SP}-1 ; m[\text{SP}] = \text{PC} ; \text{PC} = X$
- c)  $\text{SP} = \text{SP}-1 ; m[\text{SP}] = \text{PC} ; \text{PC} = \text{PC}+1$
- d)  $\text{SP} = \text{PC}-1 ; m[\text{SP}] = \text{PC} ; \text{PC} = X$

[T2.2.1]

32. Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma  $D(\text{Rb}, \text{Ri}, S)$ , sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?

- a) ESP no se puede usar como registro índice
- b) El factor de escala S puede ser 1, 2, 4, 8
- c) El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
- d) EBP no se puede usar como registro base

[T6.1]

33. El primer nivel de una jerarquía de memoria tiene una tasa de aciertos del 75% y las peticiones de memoria tardan 12 ns en completarse si dicha posición se encuentra en ese nivel y 100 ns si no es así. ¿Cuál es el tiempo medio de acceso de la jerarquía?

- a) 88 ns
- b) 112 ns
- c) 25 ns
- d) 34 ns

[T5.1]

34. ¿Qué tipo de sincronización es más conveniente en el caso de tener dispositivos con distintos requisitos de temporización?

- a) Síncrona
- b) Asíncrona
- c) No se pueden conectar dispositivos con distintos requisitos de temporización
- d) Ninguna de las anteriores

[T5.1]

35. ¿Qué técnica de E/S consume menos tiempo del procesador?

- a) E/S programada
- b) E/S mediante interrupciones
- c) E/S mediante DMA
- d) Todas pueden consumir el mismo número de recursos en función de la velocidad del dispositivo de E/S al que se acceda

[T5.1]

36. Supongamos dos CPU con idéntica anchura tanto en el bus de direcciones como en el de datos. Si una de ellas emplea E/S independiente y la otra mapeada en memoria, ¿cuál podrá acceder a una mayor cantidad de memoria?

- a) La CPU con E/S independiente
- b) La CPU con E/S mapeada en memoria
- c) Ambas podrán acceder a la misma cantidad de memoria
- d) Depende de la técnica de E/S utilizada

[T1.2]

37. ¿En qué pareja de registros están el dato/instrucción que se leerá o escribirá en memoria, y la dirección de memoria?

- a) MBR y MAR
- b) MAR y ACUMULADOR
- c) IR y ACUMULADOR
- d) MBR y PC

[T3.3]

38.. ¿Cómo actúa el indicador Z del registro de indicadores de estado?

- a) Se pone a 1 cuando el resultado es negativo
- b) Se pone a 0 cuando el resultado es negativo
- c) Se pone a 1 cuando el resultado de una operación es 0
- d) Se pone a 1 cuando el resultado es positivo

[T2.1.4]

39. ¿Cuál afirmación es FALSA al comparar las arquitecturas x86 y x86-64?

- a) El tamaño de un double es el mismo
- b) El tamaño de un puntero es el mismo
- c) El tamaño de un entero (int) es el mismo
- d) El tamaño de las posiciones de memoria es el mismo

# ENCENDER TU LLAMA CUESTA MUY POCO



[T6.1]

40. En una jerarquía de memoria, a medida que nos alejamos del procesador:

- a) el tamaño de la memoria disminuye
- b) el tamaño de la unidad de transferencia entre dos niveles aumenta
- c) el coste por byte aumenta
- d) el tiempo de transferencia disminuye

[T2.1.4]

41. Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA32 como en x86-64. Sólo una de ellas es FALSA. ¿Cuál?

- a) El tamaño de un short es 2 bytes
- b) El tamaño de un int es 4 bytes
- c) El tamaño de un double es 8 bytes
- d) El tamaño de un puntero es 4 bytes

[T5.2]

42. La consulta de estado que se puede llevar a cabo en la E/S programada sirve para...

- a) consultar si el dispositivo está dispuesto para recibir datos o tiene datos disponibles
- b) consultar qué dispositivo ha solicitado la interrupción
- c) consultar si el dispositivo funciona correctamente
- d) Ninguna de las respuestas anteriores es correcta

[T4.3]

43. ¿Cuál de las siguientes afirmaciones sobre la segmentación de cauce es cierta?

- a) En general, un operación segmentada ("pipelined") requiere el mismo tiempo o más, desde el principio hasta el fin, que la misma operación en una implementación no segmentada
- b) El CPI de un cauce superescalar es siempre 1 o menor que 1
- c) La predicción de saltos es una técnica para minimizar los riesgos de datos
- d) Un cauce ("pipeline") de instrucciones inicialmente vacío y con 3 etapas tardará siempre 5 ciclos de reloj en ejecutar 3 instrucciones si cada una de ellas utiliza las 3 etapas

[T6.2]

44. ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM dinámicas es cierta?

- a) Los datos permanecen en cada celda indefinidamente
- b) Las celdas de almacenamiento son complejas
- c) Las operaciones de escritura sirven como operaciones de refresco
- d) Las operaciones de lectura no son destructivas

[T5.1]

45. La conexión entre un dispositivo de E/S y el procesador mediante bus:

- a) Requiere mucha circuitería
- b) Permite conectar en paralelo varios dispositivos
- c) Es difícil de expandir
- d) Requiere multiplexores y demultiplexores para las señales de dato

[T2.3.1]

46. Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/IA32 almacena (reserva espacio para) una variable "var" local a una función "fun" en una dirección de memoria referenciable (en lenguaje ensamblador) como:

- a) -k(%ebp), siendo k un número constante positivo relativamente pequeño

- b)  $k(\%ebp)$ , siendo  $k$  un número constante positivo relativamente pequeño
- c)  $k(\%esp)$ , siendo  $k$  un número constante positivo relativamente pequeño
- d) var (el nombre de la variable representa su posición de memoria)

[T5.1]

47. ¿Cuál de las siguientes afirmaciones es cierta?

- a) La E/S en memoria emplea la patilla IO/M#
- b) En E/S independiente, las instrucciones de acceso a memoria suelen ser más largas que las de E/S
- c) La E/S en memoria facilita la protección
- d) Ninguna de las anteriores es cierta

[T2.1.2]

48. ¿Cuál de las siguientes afirmaciones sobre la instrucción leave es cierta?

- a) Equivale a pop %ebp seguida de mov %ebp,%esp  
en realidad es mov %ebp,%esp; pop %ebp
- b) Equivale a mov %esp,%ebp seguida de pop %ebp  
mov al revés
- c) No es obligatorio usarla. En su lugar puede realizarse una secuencia explícita de operaciones mov y pop
- d) Se ejecuta justo después de retornar de un procedimiento  
lo típico después de retornar (de una función C) es pop/add \$n,%esp y/o mov %eax,...

[T2.2.1]

49. Si el contenido de EBP es 0x13000, ¿a qué dirección se hace referencia con la instrucción INC -0x80(%EBP)?

- a) 0x12F80
- b) 0x13000
- c) 0x13080
- d) 0x80

[P2T]

50. ¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?

- a) -m elf\_i386
- b) -32
- c) -m32
- d) No hace falta modificador, ld lo deduce del tipo de objeto a enlazar

[T1.2]

51. No en todas las instrucciones máquina hay una fase de

- a) ejecución
- b) captura de operandos
- c) captación
- d) decodificación

[T1.1]

52. Un modo de vídeo de 512 x 256 píxeles y 16 colores por píxel ocupa una memoria de:

- a) 8 KB
- b) 64 KB
- c) 128 KB

d) 2 MB

[T1.3]

53. ¿Cuál de los siguientes no es un tipo de bus?

- a) Paralelo
- b) Sistema
- c) E/S
- Secuencial
  - d) opuestos a buses paralelos son los buses serie  
opuestos a programas secuenciales son los programas paralelos

[T2.1.2]

54. ¿Cuál de las siguientes afirmaciones es correcta?

- a) El lenguaje máquina es igual para todos los computadores.
- b) Las instrucciones en lenguaje ensamblador se almacenan y tratan como cadenas de unos y ceros.
- c) Las instrucciones en lenguaje máquina se almacenan y tratan como cadenas de unos y ceros.
- d) El lenguaje ensamblador es igual para todos los computadores.

[T2.2.3]

55. En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?

- mov \$-1, %edi
- a) add \$0, %edi  
add afecta
- b) mov \$-1, %edi  
mov no afecta a los flags  
sub %edi, %edi
- c) adc \$0xFFFFFFFF, %edi  
adc afecta  
mov \$0, %edi
- d) sub \$1, %edi  
sub afecta

[T2.2.1]

56. ¿Cuál de las siguientes instrucciones máquina copia en EAX la dirección efectiva resultante de la operación EDX\*4 + EBX?

- a) movl 4(%edx, %edx), %eax
- b) leal (%ebx, %edx, 4), %eax
- c) leal 4(%edx, %edx), %eax
- d) movl (%ebx, %edx, 4), %eax

## Test 2

[T2.3.2]

57. La convención de llamada Linux/GCC x86-32 considera, respecto a convenios de uso de registros:

- a) Algunos registros salva-invocante, otros salva-invocado, uno especial

- b) 8 registros salva-invocante, 6 registros salva-invocado, y 2 especiales
- c) Algunos registros para pasar argumentos, otros salva-invocante, otros salva-invocado, dos especiales
- d) 3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales

[T6.1]

58. ¿Qué tipo de localidad de las referencias a memoria se define como: "si se referencia un elemento, volverá a ser referenciado pronto"?

- a) Localidad espacial
- b) Localidad lógica
- c) Localidad temporal
- d) Ninguna de las respuestas anteriores es correcta

[T2.4.1]

59. El marco de pila en x86-64 Linux...

- se crea para funciones en las que GCC no puede evitar que RBP baje más, como por
- a) ejemplo: que haya demasiadas variables locales (y no quepan en registros), o que haya que salvar algún registro salva-invocado
- b) sólo se crea para funciones que invocan anidadamente a otra función (procedimientos padre, no hojas)
- se crea para funciones en las que GCC no puede evitar que RSP baje más, como por
- c) ejemplo: que haya que calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función
- d) no existe, porque RBP no es registro especial en x86-64

[P4T]

60. En una bomba como las estudiadas en prácticas, del tipo...

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

- a) el entero 0x804a044
- b) el entero almacenado a partir de la posición de memoria 0x804a044
- c) el string almacenado a partir de la posición de memoria 0x24(%esp)
- d) ninguna de las anteriores

[T5.1]

61. ¿Cuál de las siguientes afirmaciones es cierta?

- a) La E/S independiente facilita la protección
- b) En E/S independiente, las instrucciones de acceso a memoria se emplean tanto para memoria como para E/S
- c) La E/S en memoria es mucho más rápida que la E/S independiente

# ENCENDER TU LLAMA CUESTA MUY POCO



- d) La E/S en memoria emplea la patilla IO/M#

[T1.1]

62. En una CPU de 32 bits con memoria de bytes, el problema es que...

- a) No hay problema, cuando se salva un registro a memoria se escribe en la posición deseada
- b) Hay que usar 4 instrucciones de lectura (o escritura) para leer (o escribir) un registro completo
- c) No tiene sentido, un registro no cabría en memoria
- d) Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU

[T6.2]

63. ¿Cuál de las siguientes afirmaciones es cierta?

- a) La memoria SRAM es más lenta que la DRAM
- b) La lectura en la memoria SRAM es destructiva
- c) La memoria DRAM es más cara que la SRAM
- d) Ninguna de las anteriores

[T2.3.1]

64. ¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?

- a) Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop
- b) Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero
- c) Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable
- d) En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina

[T5.1]

65. Señale cuál de las siguientes opciones es una técnica habitual para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

- a) Acceso directo a memoria (DMA)
- b) E/S por flanco
- c) Acceso indirecto a memoria (IMA)
- d) E/S por nivel

[T2.2.5]

66. De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...

- a) El bucle do-while
- b) El bucle while
- c) La selección switch-case
- d) El bucle for

[P3T]

67. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%r"
 :[r] "+r" (result)
 :[i] "r" (i),
 [a] "r" (array));
```

¿Cuál de las siguientes afirmaciones es correcta?

- a) a es una posición de memoria de entrada
- b) i es un registro de entrada
- c) r es una posición de memoria de entrada/salida
- d) la salida de la función se fuerza a que esté en la variable result

[P2T]

68. ¿Qué modificador (switch) de gcc hace falta para compilar .c → .s (de fuente C a fuente ASM)?

- a) gcc -c
- b) gcc -s
- c) Eso no se puede hacer con gcc
- d) gcc -S

[P4T]

69. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la contraseña? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" podemos entender la cifra del César, por ejemplo. "Invertir" es darle la vuelta al string de manera que la primera letra se convierta en la última y viceversa.

- a) 1 string del usuario se cifra, y se compara con el string del fuente
- b) 2 strings del fuente se invierten, se concatenan, se cifra el resultado, y se compara con el string del usuario
- c) 2 strings del fuente se concatenan, se invierte el resultado, se cifra, y se compara con el string del usuario
- d) Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

[P1]

70. El primer parámetro de printf:

- a) es un entero
- b) es un número en coma flotante
- c) puede ser de cualquier tipo
- d) es un puntero

[P2T]

71. ¿Qué hace gcc -O?

- a) Compilar con optimización suave
- b) Compilar .c → .o (fuente C a objeto)
- c) Compilar .s → .o (fuente ASM a objeto)
- d) Ambas (b) y (c), según la extensión de los ficheros que se usen como argumentos

[T6.5]

72. Una cache de 256 B asociativa por conjuntos de 4-vías con líneas de 16 B tendría

4 conjuntos

- a)  $256\text{ B} = 2^8\text{ B} = 2^4\text{ líneas} \cdot 2^4\text{ B/línea}$   
 $2^4\text{ líneas} = 2^2\text{ conjuntos} \cdot 2^2\text{ vías}$
- b) ningún conjunto
- c) 64 conjuntos
- d) 16 conjuntos

[T2.2.1]

73. La primera letra (I) de la instrucción lea:

- a) forma parte del nemotécnico de la instrucción
- b) indica que la instrucción usa ordenación de bytes little-endian
- c) indica que la instrucción trabaja con un operando destino de 32 bits (long word)
- d) indica que la instrucción afecta a los 16 bits menos significativos del operando destino (low word)

[T3.1]

74. Respecto a MBR y MAR

- a) Ambos son accesibles por el programador ninguno
- b) Ambos permiten guardar información sobre el marco de pila sin relación directa
- c) MAR requiere menos señales de control que MBR en Tema 3 tr.10, sólo una (LoadMAR), frente a 4 de MBR (Load/Enable Mem/Bus)
- d) MAR contiene el dato/instrucción que se leerá o escribirá en memoria ese es MBR

[T3.3]

75. Sea un formato de microinstrucción que incluye dos campos independientes de 9 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorrarán en cada microinstrucción?

- a) 4
- b) 1
- c) 9
- d) 8

[T2.3.2]

76. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?

- a) EDX
- b) EAX
- c) ECX
- d) EBX

[T2.2.3]

77. En una resta de dos números en complemento a dos, se produce desbordamiento cuando...

- a) los dos operandos son negativos y el resultado es positivo.
- b) los dos operandos son positivos y el resultado es negativo.
- c) Las dos respuestas a y b son correctas.
- d) Ninguna de las anteriores es correcta.

[T1.1]

78. Son funciones de la unidad de control:

- a) la codificación de las instrucciones máquina
- b) la lectura de memoria principal de la instrucción apuntada por el µPC
- c) el secuenciamiento de las instrucciones máquina
- d) todas las respuestas son ciertas

[T1.2]

79. ¿Cuál de las siguientes afirmaciones es incorrecta?

- a) No siempre es necesario indicar la dirección de la siguiente instrucción
- b) El formato de una instrucción nos indica el significado de cada bit de la instrucción
- c) Todas las instrucciones deben tener código de operación
- d) Todas las instrucciones deben tener operando fuente y operando destino

[T5.1]

80. Ventajas de la E/S independiente (señalar la opción incorrecta)

- a) decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
- b) diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
- c) protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- d) mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)

[T6.5]

81. ¿Cuál es el tamaño de la etiqueta de cache en un ordenador capaz de direccionar por bytes 1 MB de memoria principal y 32 KB de memoria cache y correspondencia asociativa por conjuntos con 32 bytes por línea y 16 líneas por conjunto?

- a) 9 bits
- b) 6 bits
- c) 7 bits
- d) 8 bits

Entendemos "y en ese nivel de cache", no solo "en un ordenador". El ordenador no dirige 32KB de memoria cache (revisar la gramática del enunciado).

[P3T]

82. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%r"
 :[r] "+r" (result)
 :[i] "r" (i),
 [a] "r" (array));
```

¿Cuál de las siguientes afirmaciones es correcta?

- a) la salida de la función se fuerza a que esté en la variable result
- b) r es una posición de memoria de entrada/salida
- c) a es una posición de memoria de entrada
- d) i es un registro de entrada

[T6.5]

83. La política de correspondencia de una memoria cache con la mitad de conjuntos que líneas es:

Asociativa por conjuntos de 2 vías

- a) "vía" se refiere a cuántas "alternativas" (líneas) hay en un conjunto.  
Además, todas las demás opciones tienen defectos
- b) Directa con 2 líneas  
en directa no hay que decir vías ni líneas ni nada, basta saber los tamaños de línea y cache
- c) Asociativa por conjuntos con 2 líneas  
con cuántas líneas? será cuántas vías!

# ENCENDER TU LLAMA CUESTA MUY POCO



d) Totalmente asociativa de media vía

[T2.1.4]

84. ¿Cuál afirmación es FALSA en arquitecturas x86-64?

- a) El tamaño de las posiciones de memoria es 64 bits
- b) El tamaño de los registros es 64 bits
- c) El tamaño de un double es 64 bits
- d) El tamaño de un puntero es 64 bits

[T5.4]

85. ¿Cuál de los siguientes es un registro de un controlador de DMA?

- a) IR (Instruction Register)
- b) WC (Word Count)
- c) SP (Stack Pointer)
- d) PC (Program Counter)

BURN.COM

#StudyOnFire

**BURN**  
ENERGY DRINK

WUOLAH

**Nombre:**

**DNI:**

**Grupo:**

## Examen de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |

1. En la convención cdecl estándar para arquitecturas x86 de 32 bits, cuál de las siguientes afirmaciones es cierta:
  - a. Los parámetros se pasan en pila, de derecha a izquierda; es decir, primero se pasa el último parámetro, después el penúltimo... y por fin el primero.
  - b. Solamente es necesario salvar el registro EAX
  - c. Los registros EBX, ESI y EDI son salvados por el invocante
  - d. Ninguna de las anteriores es cierta

---

2. En la convención cdecl estándar para arquitecturas x86 de 32 bits, el resultado de una función se devuelve usualmente en el registro:
  - a. EBX
  - b. EBP
  - c. **EAX**
  - d. ESI

---

3. Utilizando la sentencia asm, las denominadas restricciones que se indican al final de dicha sentencia, involucran a:
  - a. Solamente las entradas
  - b. Solamente las salidas
  - c. Solamente los sobrescritos

- d. Ninguna de las anteriores es cierta

4. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a], %[i], 4), %r"
 : [r] "+r" (result)
 : [i] "r" (i),
 : [a] "r" (array));
```

¿Cuál de las siguientes afirmaciones es correcta?

- a. r es una posición de memoria de entrada/salida
- b. a es una posición de memoria de entrada
- c. i es un registro de entrada
- d. el código de retorno de la función asm se fuerza a que esté en la variable result

5. En la realización de la práctica de la bomba digital, una parte del código máquina es el siguiente:

```
0x080486e8 <main+120>: call 0x8048524 <strncmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
```

¿Cuál de las siguientes afirmaciones cambiaría el salto condicional por un salto incondicional?

- a. set \$0x080486ef=0xeb

- b. set \*(char\*)0x080486ef=0xeb
  - c. set \*(char\*)0x080486f6=jmp
  - d. set %0x080486ef=0xeb
- 

- 6. El punto de entrada de un programa ensamblador en gcc/as Linux x86 se llama
  - a. main
  - b. begin
  - c. **\_start**
  - d. \_init

---

- 7. Alguna de las siguientes líneas de código sirve para definir una variable entera llamada tam en gcc/as Linux x86. ¿Cuál?
  - a. var tam : integer;
  - b. tam: .int .-msg
  - c. \_int tam = 0
  - d. int tam;**

---

- 8. ¿Qué hace gcc -O1?
  - a. Compilar .c->.o (fuente C a objeto)
  - b. Compilar .s->.o (fuente ASM a objeto)
  - c. Ambas (a) y (b), según la extensión de los ficheros que se usen como argumentos
  - d. Compilar con optimización**

---

- 9. ¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?
  - a. **Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop**
  - b. Al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
  - c. Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero
  - d. Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable

---

- 10. Para averiguar la paridad de un número se puede usar la operación:

- a. AND
  - b. NAND
  - c. XOR**
  - d. OR
- 

- 11. En una bomba como las estudiadas en prácticas, del tipo...

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

...el código numérico (pin) es...

- a. el entero 0x804a044
  - b. el entero cuya dirección está almacenada en la posición de memoria 0x804a044
  - c. el entero almacenado a partir de la posición de memoria 0x24(%esp)
  - d. el entero almacenado a partir de la posición de memoria 0x804a044**
- 

- 12. En una bomba como las estudiadas en prácticas, del tipo...

```
0x080486e8 <main+120>: call 0x8048524 <strcmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
0x080486f6 <main+134>: ...
```

la contraseña es...

- a. el valor que tenga %eax
  - b. el string almacenado a partir de 0x80486f6
  - c. el entero almacenado a partir de donde apunta %eax
  - d. ninguna de las anteriores**
- 

- 13. La práctica "popcount" debía calcular la suma de bits de los elementos de un array.

# ¿BUSCAS LIBROS O EBOOKS SOBRE ENFERMERÍA?

AXON librería especializada en Ciencias de la Salud.

AXON  
axon.es



Encuentra todos los libros y eBooks para tu especialización, además del material complementario que necesites.

Un estudiante entrega la siguiente versión de `popcount4`:

```
int popcount4(unsigned* array, int len)
{
 int i, j, res = 0;
 for(i = 0; i < len; ++i) {
 unsigned x = array[i];
 int n = 0;
 do {
 n += x & 0x01010101L;
 x >>= 1;
 } while(x);
 for(j = 16; j == 1; j /= 2) {
 n ^= (n >>= j);
 }
 res += n & 0xff;
 }
 return res;
}
```

Esta función `popcount4`:

- a. Produce el resultado correcto
- b. Fallaría con array={0,1,2,3}
- c. Fallaría con array={1,2,4,8}
- d. No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

14. La práctica "paridad" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de `parity6`:

```
int parity6(unsigned * array, int len)
{
 int i, result = 0;
 unsigned x;
 for (i=0; i<len; i++) {
 x = array[i];
 asm("mov %[x], %%edx\n\t"
 "shr $16, %%edx\n\t"
 "shr $8, %%edx\n\t"
 "xor %%edx,%%edx\n\t"
 "setp %%dl\n\t"
 "movzx %%dl, %[x]\n\t"
 : [x] "+r" (x)
 :
 : "edx"
);
 result += x;
 }
 return result;
}
```

Esta función `parity6`:

- a. Produce el resultado correcto

- b. Fallaría con array={0,1,2,3}
- c. Fallaría con array={1,2,4,8}
- d. No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

15. ¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de una subrutina según la convención cdecl para IA32?

- a. eax
- b. ebx
- c. ecx
- d. edx

16. Sea un computador de 32 bits con una memoria caché L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:

```
int v[262144];
for (i = 0; i < 262144; i += 2)
 v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la primera ejecución del bucle anterior?

- a. 0 (ningún fallo)
- b. 1/2 (mitad aciertos, mitad fallos)
- c. 1/8 (un fallo por cada 8 accesos)
- d. 1 (todo son fallos)

17. El servidor de SWAD tiene dos procesadores Xeon E5540 con 4 núcleos cada uno. Cada procesador tiene 4 caches L1 de instrucciones de 32 KB, 4 caches L1 de datos de 32 KB, 4 caches unificadas L2 de 256 KB y una cache unificada L3 de 8MB. Suponga que un proceso swad, que se ejecuta en un núcleo, tiene que ordenar un vector de estudiantes accediendo repetidamente a sus elementos. Cada elemento es una estructura de datos de un estudiante y tiene un tamaño de 4KB. Si representamos en una gráfica las prestaciones en función del número de

estudiantes a ordenar, ¿para qué límites teóricos en el número de estudiantes se observarán saltos en las prestaciones debidos a accesos a la jerarquía de memoria?

- a. 4 / 32 / 512 estudiantes
  - b. 8 / 64 / 2048 estudiantes**
  - c. 16 / 32 / 64 estudiantes
  - d. 32 / 256 / 8192 estudiantes
- 

**18.** En la práctica de cache hemos hecho una gráfica con el código size.cc ¿Qué forma tiene la gráfica que se debe obtener?

- a. Forma de U (o V) con un tramo descendente y otro ascendente
  - b. Forma de U (o V) invertida, con un tramo ascendente y otro descendente
  - c. Forma de /, una gráfica siempre creciente y sin escalones
  - d. Una escalera con varios tramos horizontales**
- 

**19.** En la práctica de la cache, el código de line.cc incluye la sentencia

```
for (unsigned long long line=1;
line<=LINE; line<<=1) { ... }
```

¿Qué objetivo tiene la expresión line<<=1?

- a. salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
  - b. duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior**
  - c. volver al principio del vector cuando el índice exceda la longitud del vector
  - d. sacar un uno (1) por el stream line
- 

**20.** En la práctica de la cache, el código de size.cc accede al vector saltando de 64 en 64. ¿Por qué?

- a. Porque cada elemento del vector ocupa 64 bytes
- b. Para recorrer el vector más rápidamente
- c. Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB

**d. Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal**

---

**Nombre:**
**DNI:**
**Grupo:**

### Examen Test (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

1. ¿Qué salida produce el siguiente código?

Asumir representación de datos de arquitectura IA32.

```
unsigned int x = 0xDEADBEEF;
unsigned short y = 0xFFFF;
signed int z = -1;
if (x > (signed short) y)
 printf("Hello");
if (x > z)
 printf("World");
```

a. No imprime nada

b. Imprime "Hello"

c. Imprime "World"

d. Imprime "HelloWorld"

2. ¿Cuál de las siguientes instrucciones convierte %eax = 5 \* %eax?

1) mov (%eax, %eax, 4), %eax

2) lea (%eax, %eax, 4), %eax

a. Ninguna

b. Sólo 1

c. Sólo 2

d. Ambas 1 y 2

3. La instrucción test es...

a. Lo mismo que sub

b. Lo mismo que sub, pero no guarda el resultado, sólo ajusta los flags

c. Lo mismo que and

d. Lo mismo que and, pero no guarda el resultado, sólo ajusta los flags

4. En el siguiente código, ¿qué reordenamiento de los bucles muestra mejor localidad?

```
// X, Y, Z ctes #define previo
int a[X][Y][Z]
int i, j, k, sum = 0;
for (i = 0; i < Y; i++)
 for (j = 0; j < Z; j++)
 for (k = 0; k < X; k++)
 sum += a[k][i][j];
```

a. i externo, j central, k interno (el orden en que están ahora)

b. j externo, k central, i interno

c. k externo, i central, j interno

d. El orden de los bucles no afecta a la localidad

5. ¿Cuál de las siguientes expresiones toma el valor 0x01 si x es múltiplo de 32 y 0x00 en caso contrario? Asumir que x es unsigned int.

a. !(x & 0x1f)

b. !(x & 0x3f)

c. (x & 0x1f)

d. (x | 0x3f)

6. En un sistema IA32 Linux, ¿cuál es el tamaño de un long?
- 2 bytes
  - 4 bytes**
  - 6 bytes
  - 8 bytes
- 
7. Considerar la declaración C
- ```
int array[10] = {0, 1, 2, 3, 4,  
                  5, 6, 7, 8, 9};
```
- Suponer que el compilador tiene la dirección de array en el registro %ecx. ¿Cómo se movería el valor array[3] al registro %eax? Asumir que %ebx es 3.
- leal 12(%ecx),%eax
 - leal (%ecx,%ebx,4),%eax
 - movl (%ecx,%ebx,4),%eax**
 - movl 8(%ecx,%ebx,2),%eax
-
8. Alguna de las siguientes afirmaciones sobre sistemas Linux x86-64 **no** es cierta
- %rax se usa para devolver los valores de retorno de funciones
 - %eax y %ebx pueden usarse como en un sistema IA-32
 - Todos los argumentos de función se pasan a través de la pila**
 - %rbp se puede usar como cualquier otro registro (no hay puntero base)
-
9. En un sistema x86-64, si %rsp tiene el valor 0x7fffffff0000 inmediatamente antes de ejecutar una instrucción retq, ¿cuál es su valor inmediatamente después?
- 0x7fffffefff8
 - 0x7fffff0000
 - 0x7fffff0004
 - 0x7fffff0008**
-
10. Indicar cuál es la dirección de la instrucción mov en el siguiente desensamblado, donde se ha borrado parte de la dirección
- 0804xxxx: 74 12 je 08048391
0804xxxx: b8 00 00 00 00 mov \$0, %eax
-
- 08048391 + 12 = 08048403
 - 08048391 - 12 = 08048379
 - 0804837d
 - 0804837f**
-
11. En la técnica de salto retardado:
- El compilador puede reorganizar el código para llenar los huecos de retardo con instrucciones útiles.**
 - El compilador no puede insertar operaciones NOP en los huecos de retardo.
 - El salto se realiza varios ciclos antes de la instrucción de salto.
 - Las instrucciones en los huecos de retardo se ejecutan unas veces y otras no.
-
12. Si el proceso de empaquetado de una tarta de 50 segundos de duración puede segmentarse en 5 etapas, cada una de 10 segundos, de modo que 5 operarios puedan trabajar cada uno en una etapa. ¿Cuál de las siguientes afirmaciones es falsa al aplicar la segmentación?
- La ganancia de velocidad (aceleración) con la segmentación es de 5.
 - La preparación completa de cada tarta sigue requiriendo 50 s (igual que con una sola persona empaquetando las tartas).
 - Cada 50 s saldrá una nueva tarta empaquetada, el mismo tiempo que cuando no había cadena de empaquetado.**
 - Una vez en funcionamiento la segmentación, se tardará 100 segundos en tener 10 tartas empaquetadas, mientras que con un solo operario se tardaría 500 segundos.
-
13. ¿Cuál de las siguientes funciones no corresponde a la unidad de control de un procesador?
- Decodificación de las instrucciones.
 - Secuenciamiento de las instrucciones.
 - Generación de las señales de control que provocan la ejecución de cada instrucción.
 - Calculo de operaciones de coma flotante.**
-

¿BUSCAS LIBROS O EBOOKS SOBRE ENFERMERÍA?

AXON librería especializada en Ciencias de la Salud.

AXON
axon.es



Encuentra todos los libros y eBooks para tu especialización, además del material complementario que necesites.

14. Al método de interacción con los periféricos, en los que el procesador vigila periódicamente el estado de los dispositivos mediante una encuesta activa se le denomina:
- daisy-chain
 - interrupción
 - polling**
 - DMA
-
15. Se dispone de un procesador con una frecuencia de reloj de 1 GHz. Se le conecta un dispositivo que genera 100.000 interrupciones por segundo. La rutina de servicio de interrupción ejecuta 500 instrucciones. El número medio de ciclos por instrucción es 2. ¿Qué porcentaje del tiempo dedica el procesador al dispositivo?
- 1%
 - 10%**
 - 50%
 - 90%
-
16. Si el tiempo de acceso a la memoria caché es de 2 ns y el tiempo necesario para tratar un fallo de caché es de 80 ns, ¿cuál es la tasa de aciertos necesaria para que el tiempo medio de acceso al sistema de memoria sea de 10 ns?
- 0,75
 - 0,9**
 - 0,95
 - 0,8
-
17. El número -12 se almacenará en complemento a 2 en el registro %eax como:
- 0xFFFFF0C
 - 0xFF0C
 - 0xFFFFFFFF4**
 - 0xFFF4
-
18. Cuando se produce una interrupción hardware...
- Se salta al principio del programa actual.
 - Se aborta la ejecución del programa actual generando un fallo de segmentación.
-
- c. Se salta a la dirección de memoria indicada en la instrucción actual.
- d. **Se guarda el estado y se ejecuta la rutina de interrupción asociada.**
-
19. Para realizar la microoperación $\text{MAR} \leftarrow \text{PC}$, habrá que activar:
- LdPC y EnMAR
 - EnPC y LdMAR**
 - LdPC y LdMAR
 - EnPC y EnMAR
-
20. Sea un computador de 32 bits que dispone de una memoria caché de 512 KB y líneas de 64 bytes. ¿Cuántas líneas tiene la caché?
- 64
 - 1024
 - 8192**
 - 65536
-
21. Sea un computador con 48 registros y 200 instrucciones máquina. ¿Cuántas direcciones de memoria permite el formato de la instrucción de 32 bits hipotética `beqz r1, r2, dir?`
- 2^{12}**
 - 2^{14}
 - 2^{16}
 - 2^{18}
-
22. ¿Cuál es el tamaño de la marca de caché para un bus de direcciones de 48 bits (256 TB de memoria principal) y 8MB de caché L3, con un tamaño de línea de 64 B y correspondencia asociativa por conjuntos con 16 vías?
- 6 bits
 - 13 bits
 - 29 bits**
 - 48 bits
-
23. Un circuito SRAM con una capacidad de 256 Kbits tiene las patillas de direcciones A14 a A0. ¿Con cuál de las siguientes expresiones indicaría las características de

- capacidad en direcciones y datos del circuito?
- 256 K x 1
 - 64 K x 4
 - 32 K x 8**
 - 8 K x 32
-
24. En una caché con 64 bytes de longitud de línea, ¿qué bits de una dirección de memoria de 64 bits se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección? (Memoria direccionable por bytes)
- [5...0]
 - [11...6]
 - [5...3]
 - [8...6]
-
25. ¿Cuál de las siguientes afirmaciones es cierta?
- La memoria SRAM es más lenta que la DRAM.
 - La lectura en la memoria SRAM es destructiva.
 - La memoria DRAM es más cara que la SRAM.
 - Ninguna de las anteriores.**
-
26. ¿Qué conjunto de componentes permite construir una memoria 256Mx32? (sin que sobren componentes)
- 16 chips 64Mx4.
 - 32 chips 64Mx4.**
 - 16 chips 64Mx16.
 - Ninguna de las anteriores.
-
27. ¿En qué tipo de memorias coincide el tiempo de acceso y el tiempo de ciclo?
- SRAM.**
 - DRAM.
 - Tanto en a) como en b)
 - Ninguna de las anteriores.
-
28. Un procesador de 1GHz tarda 4ns en realizar 4 instrucciones sin realizar segmentación de cauce. Cuanto tardaría en realizar 9 instrucciones una versión de dicho procesador con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones.
- 2 ns.
 - 3 ns.**
 - 4.5 ns.
 - 9 ns.
-
29. ¿Cuál es el ancho del bus de direcciones para una memoria DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?
- 20
 - 16
 - 30
 - 15**
-
30. ¿Cuál de las siguientes afirmaciones es cierta?
- Al realizar la segmentación de cauce aumenta en general el tiempo necesario para la ejecución de un programa
 - Debido a que pueden existir dependencia de datos, los resultados de un programa pueden ser diferentes a si el programa se ejecutara sin segmentación.
 - La segmentación de cauce disminuye el número de instrucciones necesarias para la ejecución de un programa.
 - Ninguna de las afirmaciones anteriores.**

Nombre:

DNI:

Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿Cuál es el valor mínimo (más negativo) que puede tomar un entero de 32bits en complemento a dos?
 - a. -2^{32}
 - b. $-2^{32} + 1$
 - c. -2^{31}
 - d. $-2^{31} + 1$

2. Después de ejecutar el siguiente código, ¿qué variables serán igual a 0? (Suponer ints de 32bits y longs de 64bits)


```
unsigned int a = 0xffffffff;
unsigned int b = 1;
unsigned int c = a + b;
unsigned long d = a + b;
unsigned long e =
    (unsigned long)a + b;
```

 - a. Ninguna
 - b. c
 - c. c y d
 - d. c, d, y e

3. En la práctica 2 se pide sumar una lista de 32 enteros SIN signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando acarreos. ¿Cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32bits (sin signo)?
 - a. 0xfc00 0000
 - b. 0xfbff ffff
 - c. 0x0800 0000
 - d. 0x07ff ffff

4. En la práctica 2 se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32bits (en complemento a 2)?
 - a. 0xfc00 0000
 - b. 0xfbff ffff
 - c. 0xf800 0000
 - d. 0xf800 0001

5. En la práctica 2 se pide calcular la media y resto de una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. ¿Qué (media : resto) se debe obtener para una lista rellena a -1 salvo el primer elemento, que valiera -31?
 - a. (-1 :-30)
 - b. (-1 :-31)
 - c. (-2 : 1)
 - d. (-2 : 2)

6. ¿Cuál es el resultado de evaluar la expresión $1110_2 \wedge 1010_2$ en lenguaje C?
 - a. 1111_2
 - b. 1010_2
 - c. 0110_2
 - d. 0100_2



Tienda 4

Tienda 3

Tienda 2

Tienda 1

7. En la práctica 3 se pide calcular la suma de paridades de una lista de enteros sin signo. Suponer que un estudiante entrega la siguiente versión

```
int paridad5(unsigned* array,
              int len) {
    int i, k, result = 0;
    unsigned x;
    for (i = 0; i < len; i++) {
        x = array[i];
        for (k = 16; k == 1; k /= 2)
            x ^= x >> k;
        result += (x & 0x01);
    }
    return result;
}
```

Esta función:

- a. es correcta
 - b. falla para **array={0,1,2,3}**
 - c. **falla para array={1,2,3,4}**
 - d. no se puede marcar una y sólo una de las opciones anteriores
-

8. Suponer una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponer que para la siguiente secuencia de direcciones enviadas a la cache: 0, 2, 4, 8, 16, 32, la tasa de acierto es 0.33. ¿Cuál es el tamaño de bloque de la cache?

- a. 4 bytes
 - b. **8 bytes**
 - c. 16 bytes
 - d. Ninguno de los anteriores
-

9. Abajo se ofrece el listado de una función para multiplicar matrices $C = A \times B$.

```
void mult_matr(    float A[N][N],
                  float B[N][N], float C[N][N]) {
    /* Se asume valor inicial C = {0,0,...} */
    int i,j,k;
    for (i=0; i<N; i++)
        for (j=0; j<N; j++)
            for (k=0; k<N; k++)
                C[i][j] += A[i][k] * B[k][j];
}
```

Suponer que:

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.

- N es grande, una fila o columna no cabe completa en cache.

- El tamaño de los tipos de datos es como en IA32.

- El compilador optimiza el acceso a $C[i][j]$ en un registro.

Aproximadamente, ¿qué tasa de fallos se podría esperar de esta función para valores grandes de N?

- a. 1/16
 - b. 1/8
 - c. 1/4
 - d. 1/2**
-

10. Con los mismos supuestos, imaginar que se modifica la última sentencia (el cuerpo anidado) por esta otra

```
C[i][j] += A[i][k] * B[j][k];
```

de manera que se calcule $C = A \times B'$ (A por traspuesta de B). Aproximadamente, ¿qué tasa de fallos se podría esperar de esta nueva función para valores grandes de N?

- a. 1/16**
 - b. 1/8
 - c. 1/4
 - d. 1/2
-

11. La instrucción necesaria para cargar 0x07 en %eax es:

- a. **movb \$0x07,%eax**
 - b. **movl \$0x07,%eax**
 - c. **movb \$0x07,%al**
 - d. **movl \$0x07,%ah**
-

12. La(s) instrucción(es) necesaria(s) para cargar el dividendo 0xa30bf18a en la pareja edx:eax como paso previo a una división sin signo son:

- a. **movl \$0xf18a,%eax**
 - movl \$0xa30b,%edx**
 - b. **movl \$0xa30bf18a,%eax**
 - xorl %edx,%edx**
 - c. **movq \$0xa30bf18a,%rax**
 - d. **movl \$0xa30bf18a,%eax**
 - cltd**
-

13. ¿Cuál es el efecto de la instrucción siguiente?

mov 8(%ebp), %ecx

- a. Suma 8 al contenido de ebp y almacena la suma en ecx
- b. **Suma 8 al contenido de ebp, trata la suma como una dirección de memoria y almacena el contenido de esa dirección en ecx**
- c. Suma 8 al contenido de la posición de memoria cuya dirección está almacenada en ebp, y almacena la suma en ecx
- d. Suma los contenidos de ebp y de la dirección de memoria 8 y almacena la suma en ecx

14. Si el registro eax contiene el siguiente valor binario:

31	2423	1615	8	7	0
11111111	10101010	01010101	11110000		

¿Cuál será el valor de %eax tras ejecutar la instrucción **xorb %al, %al**?

- a. 11111111 10101010 01010101 11110000
- b. 00000000 10101010 01010101 11110000
- c. 00000000 00000000 00000000 00000000
- d. 11111111 10101010 01010101 00000000**

15. ¿Qué valor contendrá el registro edx tras ejecutar las dos instrucciones siguientes?

```
movl $-1, %edx  
movb $1, %dl
```

- a. 11111111 11111111 11111111 11111111
- b. 11111111 11111111 11111111 00000001**
- c. 00000000 00000000 00000000 00000001
- d. 00000001 00000000 00000000 00000000

16. ¿Qué valor contendrá edx tras ejecutar las siguientes instrucciones?

```
xor %eax, %eax  
sub $1, %eax  
cltd  
idiv %eax
```

- a. 0**
- b. 1
- c. -1
- d. no puede saberse con los datos del enunciado

17. ¿Cuál de los siguientes fragmentos es correcto para comenzar un programa en ensamblador que conste de un solo archivo .s?

```
a. .text  
_start:  
b. .section .text  
.local _start  
_start:  
c. .section .text  
.global _start  
_start:  
d. section .text  
.start _global  
_start:
```

18. En un programa en ensamblador queremos crear espacio para una variable entera var inicializada a 1. La línea que hemos de escribir en la sección de datos es:

```
a. .int var 1  
b. var: .int 1  
c. .int: var 1  
d. int var 1
```

19. ¿Cuántos operandos acompañan a la instrucción PUSH en IA32?

- a. 0
- b. 1**
- c. 2
- d. 3

20. ¿Para qué se utiliza la función gettimeofday en la práctica de la "bomba digital"?

- a. Para cronometrar y poder comparar las duraciones de las distintas soluciones del programa.
- b. Para imprimir la hora en la pantalla.
- c. Para cifrar la contraseña en función de la hora actual.
- d. Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la contraseña o el PIN.**

Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Cuál fue el primer procesador de Intel de 64-bits en la familia x86(-64)?
 - a. 8086
 - b. 386
 - c. **Pentium 4F**
 - d. Core i7

2. ¿Cuál es la diferencia entre las instrucciones **mov** y **lea**?
 - a. **lea** referencia (accede) la posición indicada, mientras que **mov** no lo hace.
 - b. **mov** referencia (accede) la posición indicada, mientras que **lea** no lo hace.
 - c. **lea** puede usarse para copiar un registro a otro, mientras que **mov** no.
 - d. **mov** puede usarse para copiar un registro a otro, mientras que **lea** no.

3. Al llamar a una función de 2 argumentos **foo(arg1, arg2)**, ¿cuál es el orden correcto de las operaciones? (suponiendo convención de llamada x86 cdecl, y que **foo** requiere ajustar marco de pila, esto es, salvar %ebp)
 - a. push arg1, push arg2, call foo, push %ebp
 - b. push arg1, push arg2, push %ebp, call foo
 - c. **push arg2, push arg1, call foo, push %ebp**

- d. **push arg2, push arg1, push %ebp, call foo**

4. Considere una función C declarada así:


```
void fun4arg(int a, int b, int c, int d);
```

 Suponiendo que **fun4arg** se ha compilado para una máquina x86 IA32 con enteros de 4 bytes, ¿cuál sería la dirección del argumento **b** relativa a %ebp, en el marco de pila de **fun4arg**?
 - a. %ebp + 8
 - b. **%ebp + 12**
 - c. %ebp + 16
 - d. %ebp + 20

5. ¿Cuál de las siguientes afirmaciones NO es cierta? (entender que x86=IA32)
 - a. x86-64 proporciona un espacio de memoria virtual mayor que x86
 - b. Las disciplinas de pila para x86 y x86-64 son diferentes
 - c. x86 usa %ebp como puntero base para el marco de pila
 - d. **x86-64 usa %rbp como puntero base para el marco de pila**

6. Alguna de las siguientes NO es una operación básica de la unidad de control:

- a. Transferir un registro a otro
- b. (Leer / escribir) un registro (de / a) memoria
- c. (Guardar/recuperar) registro (en/de) la pila
- d. Realizar operación ALU y guardar resultado en registro
-
7. Alguna de las siguientes señales NO es salida de la unidad de control. ¿Cuál?
- a. Dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)
- b. Señales de carga, habilitación y/o desplazamiento de registros (Load, Enable, ShiftL, ShiftR)
- c. Códigos de selección en multiplexores, decodificadores, ALU, etc (00,01,10,11...)
- d. Señales de lectura y escritura en memoria (RD, WR)
-
8. Alguno de los siguientes NO es un motivo de que no se alcance la ganancia ideal en un cauce segmentado
- a. El propio coste de la segmentación (carga de los registros de acople, etc...)
- b. La duración del ciclo de reloj impuesta por la etapa más lenta
- c. Los riesgos (hazards)
- d. La emisión múltiple (y posiblemente desordenada) de instrucciones
-
9. Alguna de las siguientes NO es una ventaja de la E/S independiente (separada, aislada)
- a. Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
- b. Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- c. Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- d. Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
-
10. Alguna de las siguientes NO es una técnica de E/S de las estudiadas en clase:
- a. E/S programada
- b. E/S asíncrona
- c. E/S controlada por interrupciones
- d. E/S mediante Acceso Directo a Memoria
-
11. Alguna de las siguientes técnicas NO es de utilidad para determinar la causa de una interrupción
- a. Múltiples líneas de interrupción INT1#, INT2#...
- b. Línea de reconocimiento INTA#
- c. Consulta de estado, o polling
- d. Interrupciones vectorizadas
-
12. Sea una cache asociativa por conjuntos de 4-vías. ¿Cuál de las siguientes afirmaciones es cierta?
- a. La cache tiene 4 bloques por línea
- b. La cache tiene 4 conjuntos por línea
- c. La cache tiene 4 líneas por conjunto
- d. La cache tiene 4 conjuntos por bloque
-
13. ¿Cuál de las siguientes afirmaciones sobre memorias cache es cierta? Recordar que llamamos "latencia" al tiempo transcurrido desde que se envía una dirección a la cache hasta que se obtiene el dato (suponiendo que se trata de un acierto), y depende por tanto del tiempo que emplee la circuitería hardware en obtenerlo. Suponer similares condiciones para ambas caches (tamaño de bloque, tamaño total, tecnología, frecuencia...)
- a. Las caches totalmente asociativas ofrecen mejor latencia, mientras que las de correspondencia directa tienen tasas de fallo más bajas
- b. Las caches totalmente asociativas ofrecen tasas de fallo más bajas, mientras que las de correspondencia directa tienen mejor latencia
- c. Las caches de correspondencia directa tienen mejor latencia y tasa de fallos
- d. Ambas tienen en general similar latencia y tasa de fallos
-
14. Un programa de ordenador que convierte un programa fuente de alto nivel completo en lenguaje máquina se llama un:



Tienda 4

Tienda 3

Tienda 2

Tienda 1

- a. intérprete
 - b. simulador
 - c. compilador**
 - d. ensamblador
-
15. En IA32 la pila es:
- a. un registro de 16 bits en el microprocesador
 - b. un registro de 32 bits en el microprocesador
 - c. una dirección de memoria de 32 bits almacenada en el contador de programa
 - d. un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa**
-
16. Cuando se ejecuta la instrucción **ret** al final de una subrutina:
- a. la dirección de comienzo de la pila se transfiere al puntero de pila
 - b. la dirección de memoria de la instrucción **ret** se transfiere al contador de programa
 - c. la dirección almacenada en la cima de la pila se transfiere al contador de programa**
 - d. la dirección almacenada en la cima de la pila se transfiere al puntero de pila
-
17. En el direccionamiento inmediato el operando reside en:
- a. en un registro del procesador
 - b. en la instrucción tras el código de operación**
 - c. en memoria, en la dirección indicada
 - d. en la pila
-
18. Un overflow nunca puede ocurrir cuando:
- a. se suman dos números positivos
 - b. se suman dos números negativos
 - c. se suma un número positivo a un número negativo**
 - d. se resta un número positivo de un número negativo
-
19. Despues de ejecutar una instrucción de suma sobre dos números con signo de la que sabemos que no provocará overflow (los dos números son pequeños en valor absoluto), queremos comprobar si el resultado de la

suma es menor que 0. ¿Qué flag necesita comprobar la instrucción de salto condicional equivalente a... ?

if (resultado<0) then goto label

- a. CF
 - b. OF
 - c. SF**
 - d. ZF
-

20. Al ejecutar el fragmento de código:

```
leal -48(%eax), %edx
cmpb $9, %edx
ja .L2
```

se salta a .L2 si el contenido del registro **%eax**:

- a. está dentro del intervalo [48,57]
 - b. es mayor o igual que 48
 - c. es mayor o igual que 57
 - d. está fuera del intervalo [48,57]**
-

21. La instrucción **cmoveb %edx, %eax**

- a. Copia en %eax el contenido de %edx si el indicador de acarreo es 1**
 - b. Copia el byte bajo de %edx en el byte bajo de %eax
 - c. Copia en %eax el byte de memoria apuntado por la dirección contenida en %edx
 - d. Copia en %eax el contenido de %edx si %eax es menor que %edx
-

22. Cuando un Intel Atom ejecuta una instrucción CALL

- a. el registro eip se guarda en la pila**
 - b. los registros eip, ebp y esp se guardan en la pila
 - c. los registros eax, ecx, edx y eip se guardan en la pila
 - d. no se guarda ningún registro
-

23. ¿Cuál de las siguientes funciones es una tarea propia de la unidad de control en la CPU?

- a. almacenar datos del programa

- b. almacenar instrucciones del programa
c. realizar operaciones lógicas
d. decodificar las instrucciones del programa
-
- 24.** ¿Cuál de los siguientes registros se utiliza para guardar la dirección de memoria donde se localiza la instrucción siguiente?
- a. **Program Counter**
b. Memory Address Register
c. Memory Data Register
d. Instruction Register
-
- 25.** ¿Cuál de las siguientes afirmaciones sobre la segmentación de cauce es cierta?
- a. La predicción de saltos es una técnica para minimizar los riesgos de datos
b. El CPI de un cauce superescalar es siempre 1 o menor que 1
c. En general, una operación segmentada ("pipelined") requiere el mismo tiempo o más, desde el principio hasta el fin, que la misma operación en una implementación no segmentada
d. Un cauce ("pipeline") de instrucciones inicialmente vacío y con 3 etapas tardará siempre 5 ciclos de reloj en ejecutar 3 instrucciones si cada una de ellas utiliza las 3 etapas.
-
- 26.** Las interrupciones generadas por el teclado interrumpirán al procesador:
- a. sólo cuando el procesador no esté realizando un trabajo útil
b. sólo si el procesador tiene activado el indicador de habilitación de interrupciones
c. sólo si el procesador está chequeando el estado del teclado
d. siempre que el usuario pulse una tecla en el teclado
-
- 27.** Las interrupciones iniciadas por un dispositivo de E/S son normalmente:
- a. internas
b. externas
c. espurias
d. software
-

28. ¿Cuántas líneas de dirección (patillas) son necesarias para direccionar un chip de memoria DRAM de 4096×4 ?

- a. 10
b. 6
c. 11
d. 12

29. Suponga el siguiente diseño caché/memoria: direcciones de 16 bits, direccionamiento por bytes, tamaño de caché 256 bytes, tamaño de bloque 8 bytes, tamaño de etiqueta 11 bits. ¿Cuál es la asociatividad (cuántas vías hay en la caché)?

- a. 8**
b. 2
c. 5
d. 4

30. Considere un sistema de memoria para un procesador de 32 bits con cachés separadas para código y datos. Suponga que el procesador direcciona la memoria por bytes y realiza accesos a palabras de 32 bits y que el espacio de direcciones es de 2^{32} bytes. La caché de datos tiene las siguientes características: 64 KB de capacidad, asociativa por conjuntos con 2 vías, y bloques de 2 palabras. ¿Cuántos bits tiene el campo etiqueta de una dirección de memoria?

- a. 11
b. 13
c. 15
d. 17

Nombre:
DNI:
Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿De qué tipo son los procesadores Intel que usamos en los laboratorios?
 - a. little-endian
 - b. big-endian
 - c. puede ajustarse mediante un bit de control en el registro CR0 que funcionen como little- o big-endian
 - d. el concepto de endian no es aplicable a estas máquinas, ya que un registro del procesador no cabe en una posición de memoria

 2. ¿En qué registro pasa el primer argumento a una función según el estándar _cdecl en una arquitectura IA32?
 - a. edi
 - b. esi
 - c. eax
 - d. Las anteriores respuestas son erróneas**

 3. ¿Cuál de las siguientes instrucciones es incorrecta en ensamblador GNU/as IA32?
 - a. pop %eip**
 - b. pop %ebp
 - c. mov (%esp),%ebp
 - d. lea 0x10(%esp),%ebp

 4. En un sistema Linux x86-64, ¿cuál de las siguientes variables ocupa más bytes en memoria?
 - a. 0xABFC
- a. char a[7]
 - b. short b[3]
 - c. int *c**
 - d. float d
-
5. Si EBX=0x00002a00 y EDI=0x0000000a, ¿cuál es la dirección efectiva en la instrucción add 0x64(%ebx,%edi,2),%eax?
 - a. 0x00002a70
 - b. 0x00002a78**
 - c. 0x00005478
 - d. 0x000054dc

 6. Las instrucciones que asignan la dirección 0x78e00 a un puntero situado en la posición 0xff00 son:
 - a. mov 0x78e00,%eax; mov %eax,0xff00
 - b. mov 0x78e00,%eax; lea %eax,0xff00
 - c. mov \$0x78e00,%eax; mov %eax,0xff00**
 - d. mov \$0x78e00,%eax; lea %eax,0xff00

 7. Si ECX vale 0, la instrucción adc \$-1,%ecx
 - a. Pone CF=1
 - b. Pone CF=0
 - c. Cambia CF
 - d. No cambia CF**

 8. Si el contenido de ESP es 0xAC00 antes de ejecutar la instrucción push %ebx. ¿Cuál será su contenido tras ejecutarla?
 - a. 0xABFC

- b. 0xABFE
- c. 0xAC02
- d. 0xAC04

9. En IA32, tras dividir 0x640000 (64 bits) entre 0x8000 (32 bits), el resultado será:

- a. 0x0 en AX y 0xC8 en DX.
- b. 0x0 en EAX y 0xC8 en EDX.
- c. 0xC8 en AX y 0xC8 en DX.
- d. 0xC8 en EAX y 0x0 en EDX.**

10. Si los contenidos de ESP y EBP son, respectivamente, 0x0008d040 y 0x00000000 antes de ejecutar una instrucción call, tras ejecutar el ret correspondiente, los contenidos serán:

- a. 0x0008d040 y 0x00000000**
- b. 0x0008d040 y 0x0008d040
- c. 0x0008d03c y 0x0008d03c
- d. 0x0008d044 y 0x00000000

11. La instrucción leave hace exactamente lo mismo que

- a. mov %ebp, %esp; pop %ebp**
- b. pop %eip
- c. mov %esp, %ebp; pop %esp
- d. ret

12. Sean un int*a y un int n. Si el valor de %ecx es a y el valor de %edx es n, ¿cuál de los siguientes fragmentos de ensamblador se corresponde mejor con la sentencia C return a[n]?

- a. ret (%ecx,%edx,4)
- b. leal (%ecx,%edx,4),%eax; ret
- c. mov (%ecx,%edx,4),%eax; ret**
- d. mov (%ecx,%edx,1),%eax; ret

13. ¿Cuál de los siguientes fragmentos es correcto para comenzar un programa en ensamblador que conste de un solo archivo .s?

- a. **.text**
_start:
- b. .section .text**
.global _start
_start:

c. **.section .text**
.local _start
_start:

d. **section .text**
.start _global
_start:

14. En la práctica 2 se pide sumar una lista de 32 enteros SIN signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando acarreos. ¿Cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32bits (sin signo)?

- a. **0x07ff ffff**
- b. 0x0800 0000**
- c. **0xfbff ffff**
- d. **0xfc00 0000**

15. La práctica "paridad" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned * array,
            int len) {
    int i, result = 0;
    unsigned x;
    for (i=0; i<len; i++) {
        x = array[i];
        asm("mov %[x], %%edx\n\t"
            "shr $16, %%edx\n\t"
            "shr $8, %%edx\n\t"
            "xor %%edx,%%edx\n\t"
            "setpe %%dl\n\t"
            "movzx %%dl, %[x]\n\t"
            : [x] "+r" (x)
            :
            : "edx"
            );
        result += x;
    }
    return result;
}
```

Esta función parity6:

- a. Produce el resultado correcto
- b. No es correcta, fallaría por ejemplo con array={0,1,2,3}**
- c. No es correcta, fallaría por ejemplo con array={1,2,4,8}

- d. No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

16. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%r"
    :[r] "+r" (result)
    :[i] "r" (i),
    [a] "r" (array));
```

¿Cuál de las siguientes afirmaciones es incorrecta?

- a. r es un registro de entrada/salida
- b. a es un registro de entrada
- c. i es un registro de salida
- d. se desea que el valor calculado por la instrucción ensamblador quede almacenado en la variable C result

17. En una bomba como las estudiadas en prácticas, del tipo...

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

- a. el entero 0x804a044
- b. el entero almacenado a partir de la posición de memoria 0x804a044
- c. el string almacenado a partir de la posición de memoria 0x24(%esp)
- d. ninguna de las anteriores

18. En la práctica de la cache, el código de line.cc incluye la sentencia

```
for (unsigned long long line=1;
line<=LINE; line<<=1) { ... }
```

¿Qué objetivo tiene la expresión line<<=1?

- a. salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
- b. duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior
- c. volver al principio del vector cuando el índice exceda la longitud del vector
- d. sacar un uno (1) por el stream line

19. En la práctica de la cache, el código de size.cc accede al vector saltando de 64 en 64. ¿Por qué?

- a. Porque cada elemento del vector ocupa 64 bytes
- b. Para recorrer el vector más rápidamente
- c. Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB
- d. Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal

20. En el programa size de la práctica de la cache, si el primer escalón pasa de tiempo=2 para un tamaño de vector menor que 32KB a tiempo=8 para un tamaño mayor que 32KB, podemos asegurar que:

- a. La cache L1 es cuatro veces más rápida que la cache L2.
- b. La cache L1 es seis veces más rápida que la cache L2.
- c. La cache L1 es al menos cuatro veces más rápida que la cache L2.
- d. La cache L1 es como mucho cuatro veces más rápida que la cache L2.

Nombre:	
DNI:	

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Por qué se impusieron las arquitecturas de registros de propósito general a las arquitecturas basadas en pila?

- a. Porque no se puede programar una arquitectura de pila en un lenguaje de alto nivel.
- b. Porque las basadas en registros son capaces de lograr un mejor rendimiento cuando se asignan variables a registros.
- c. Porque la memoria es más cara que los registros.
- d. Porque las basadas en registros permiten reducir el tamaño del programa.

2. ¿Cuál de las siguientes afirmaciones es incorrecta?

- a. El direccionamiento indexado es útil para manejo de vectores.
- b. El direccionamiento indirecto indica la dirección del operando.
- c. En el direccionamiento implícito no se indica la ubicación del operando
- d. En el direccionamiento inmediato el dato se encuentra en la propia instrucción

3. ¿Cuál de las siguientes afirmaciones sobre el direccionamiento absoluto es falsa?

- a. La instrucción contiene la dirección de memoria en la que se encuentra el objeto.
- b. El objeto está en una posición de la memoria.
- c. El rango de posiciones direccionables queda limitado por el tamaño del campo de direccionamiento.

d. El tamaño del operando direccionado queda limitado por el nº de bits del campo de direccionamiento.

-
- 4. ¿Cuál es la diferencia entre los desplazamientos a la derecha lógico y aritmético?
 - a. El lógico inserta siempre ceros en el bit más a la derecha
 - b. El aritmético inserta en el bit más a la derecha una copia del bit de signo
 - c. Insertan de forma distinta el bit más a la izquierda
 - d. Ninguna, la diferencia es entre los desplazamientos a la izquierda

5. La instrucción seta %al (seta significa set if above):

- a. Pone AL a 1 si CF=0 y ZF=0
- b. Pone AL a 1 si CF=0 o ZF=0
- c. Pone AL a 1 si CF=1 y ZF=0
- d. Pone AL a 1 si CF=1 o ZF=1

6. Los arrays bidimensionales en lenguaje C se almacenan en orden...

- a. "mayor-de-columna" (column-major)
- b. "mayor-de-fila" (row-major)
- c. "de mayor a menor" (major-to-minor)
- d. "de fila a columna" (file-to-column)

7. ¿Cuál sería el "equivalente x86-64" del "pseudo-código C" rcx = ((int*)rax)[rcx]?

- a. mov (%rax,%rcx,4),%rcx
- b. lea (%rax,%rcx,4),%rcx
- c. lea (%rax,4,%rcx),%rcx

d. `mov (%rax,4,%rcx),%rcx`

8. El direccionamiento directo a memoria utiliza

- a. un registro.
 - b. dos desplazamientos contenidos en la propia instrucción.
 - c. un registro y un desplazamiento contenidos en la propia instrucción.
 - d. un desplazamiento.**
-

9. ¿Cuál de las siguientes afirmaciones es falsa?

- a. Los programas necesitan una forma de pasar parámetros a las subrutinas y de recibir las salidas de vuelta.
 - b. Las subrutinas necesitan ser capaces de reservar espacio en memoria para las variables locales sin sobrescribir ningún dato usado por el programa que hace la llamada.
 - c. Las subrutinas necesitan recibir parámetros desde el programa que hace la llamada que indiquen qué registros pueden alterar y cuáles no.**
 - d. Las subrutinas necesitan algún modo de saber desde dónde han sido llamadas para poder volver al programa que realizó la llamada cuando se completa la subrutina.
-

10. Respecto a la segmentación:

- a. Cuanto más parecidos sean el tiempo de ejecución de una instrucción sin segmentar y el tiempo de una etapa en el procesador segmentado, mayor será la ganancia máxima que se puede obtener.
 - b. Cuando el número de instrucciones ejecutadas tiende al número de etapas de un procesador segmentado, la ganancia máxima que se puede obtener tiende a infinito.
 - c. Cuanto mayor sea la relación entre el tiempo de ejecución de una instrucción sin segmentar y el tiempo de una etapa en el procesador segmentado, mayor será la ganancia máxima que se puede obtener.**
 - d. Cuando el número de instrucciones ejecutadas en un procesador segmentado crece, la ganancia máxima que se puede obtener tiende a 1
-

11. Respecto a la segmentación, ¿cuál de las siguientes afirmaciones es falsa?

- a. La técnica de register forwarding habilita una serie de caminos (buses) que se añaden al cauce para permitir que los resultados de una etapa pasen como entradas a la etapa donde son necesarias.
 - b. La reorganización del código y la introducción de instrucciones `nop` permite evitar dependencias de datos.
 - c. Retrasar la fase de decisión saltar/no saltar de las instrucciones de salto condicional contribuye a mejorar el rendimiento del procesador.**
 - d. Cuantas más etapas tenga un cauce, más instrucciones se estarán ejecutando en distintas fases y más posibilidades se presentan de que existan riesgos entre ellas.
-

12. La precaptación (cola de instrucciones) está relacionada con...

- a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)**
 - b. Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
 - c. Los riesgos de control (intenta determinar de antemano el flujo de control)
 - d. Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)
-

13. La predicción de saltos está relacionada con...

- a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
 - b. Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
 - c. Los riesgos de control (intenta determinar de antemano el flujo de control)**
 - d. Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)
-

14. ¿Cuál de los siguientes modos de direccionamiento es *menos* preferible para un procesador con segmentación de cauce?

- a. Registro
- b. Indirecto a través de memoria**
- c. Indirecto a través de registro

d. Indexado (o relativo a base, o base+índice)

15. Si se dice que en un sistema computador cada dirección especifica uno o dos puertos de E/S, se refiere a que:

- a. La misma dirección (por ejemplo 0x0210) puede ser una posición de memoria o un puerto de E/S, según IO/M#
- b. La misma dirección puede usarse para transferir un byte o una palabra de mayor tamaño (ese byte y el siguiente)
- c. **Un puerto será de sólo lectura, otro de sólo escritura, y ambos se decodifican en la misma dirección**
- d. La pregunta es capciosa, una dirección puede especificar un puerto, no dos

16. Respecto a si un computador dispone de E/S independiente (separada) o usa E/S mapeada a memoria:

- a. Si el encapsulado (chip) del procesador tiene patilla (pin) IO/M# (o patillas equivalentes), eso evidencia que el computador usa E/S mapeada a memoria
- b. **Si el repertorio del procesador tiene instrucciones del tipo IN y OUT, es que el computador dispone de E/S separada**
- c. Si el encapsulado del procesador no dispone de patilla IO/M# (ni equivalentes), el computador sólo dispone de E/S separada
- d. Si el repertorio del procesador tiene instrucciones del tipo LOAD y STORE, el computador sólo dispone de E/S mapeada a memoria

17. Supongamos dos procesadores con bus de direcciones con idéntico número de líneas. Si uno de ellos emplea E/S mapeada en memoria y el otro E/S independiente, ¿cuál podrá acceder a una mayor cantidad de memoria?

- a. **El que tiene E/S independiente.**
- b. El que tiene E/S mapeada en memoria.
- c. Depende del tamaño del bus de direcciones.
- d. Ambos podrán acceder a la misma cantidad de memoria.

18. ¿Cuál de las siguientes afirmaciones es incorrecta?

- a. La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que

siempre están listos para transferir) y con E/S por IRQ (cuando se usa polling para determinar el origen de la IRQ)

- b. Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó)
- c. **Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S**
- d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S

19. El fragmento de código:

```
poll: in a, 0x20  
      cmp a, $0  
      jnz poll  
      load a, 0x11  
      out 0x21, a
```

corresponde a:

- a. E/S programada sin consulta de estado
- b. **E/S programada con consulta de estado**
- c. E/S por interrupciones
- d. E/S por DMA

20. Un procesador con E/S en el mapa de memoria tiene un bus de direcciones de 10 líneas y uno de datos de 8. El mapa de memoria tiene 512 posiciones para instrucciones, 256 para datos y 256 para E/S, en ese orden. Los rangos de direcciones para esas tres zonas serán:

- a. **000 a 1FF, 200 a 2FF y 300 a 3FF.**
- b. 000 a 5FF, 600 a 7FF y 800 a 9FF.
- c. 000 a 7FF, 800 a BFF y C00 a FFF.
- d. 000 a 9FF, A00 a CFF y D00 a FFF.

21. El instrumento GIADA de la sonda espacial

ROSETTA (diseñado en Granada) está basado en un microprocesador 8086 y el siguiente mapa de memoria:

RAM volátil: 00000 - 0FFFF

RAM no volátil: 10000 - 1FFFF

ROM: F0000 - FFFFF

¿Cuál es el tamaño total de la memoria?

- a. 48KB
- b. **192KB**
- c. 2MB

d. 3MB

22. Una memoria que está estructurada en palabras de 8 bits tiene una capacidad de 64 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?

- a. 8
 - b. 12
 - c. 13
 - d. 24
-

23. ¿Cuál de los siguientes es el ejemplo más acertado de localidad espacial?

- a. Referenciar elementos de un array sucesivamente
 - b. Iterar repetidamente el cuerpo de un bucle
 - c. Reservar dinámicamente (malloc) espacio para una estructura o union
 - d. Referenciar continuamente la misma variable local
-

24. ¿Cuál de las siguientes es una idea fundamental de la jerarquía de memoria?

- a. Crear una gran cantidad de almacenamiento que sea caro y rápido
 - b. Crear una pequeña cantidad de almacenamiento que sea caro y lento
 - c. Que dispositivos más pequeños y rápidos sirvan de cache para dispositivos más grandes y lentos
 - d. Que dispositivos más grandes y lentos sirvan de cache para dispositivos más pequeños y rápidos
-

25. ¿Cuál de las siguientes afirmaciones acerca de las memorias SRAM es falsa?

- a. Los datos almacenados se mantienen por un tiempo indefinido mientras se mantenga la alimentación.
 - b. El número de transistores necesario para implementar cada celda es mayor que en las memorias RAM dinámicas
 - c. Las operaciones de lectura son destructivas
 - d. Son más veloces que las memorias RAM dinámicas
-

26. Se dispone de un circuito integrado que actúa como módulo básico de memoria de $8K \times 4$. ¿Qué circuitos necesitamos para construir una memoria de $16K \times 8$?

- a. 2 módulos de memoria.

b. 2 módulos de memoria y un decodificador.

c. 4 módulos de memoria.

d. 4 módulos de memoria y un inversor.

27. En la cache L1 de instrucciones, la tasa de fallos:

- a. Siempre tiende a crecer si el tamaño total de L1 crece.
 - b. Siempre tiende a disminuir si el tamaño total de L1 crece.
 - c. Siempre tiende a crecer si el número de vías crece.
 - d. Siempre tiende a disminuir si el número de vías disminuye.
-

28. Supongamos que una memoria cache de tamaño 512B con política LRU está inicialmente vacía. Si para la siguiente secuencia de direcciones enviadas a la cache: 0, 2, 4, 8, 16, 32, la tasa de acierto es 0.33, ¿cuál es entonces el tamaño de bloque de la cache?

- a. 4B
 - b. 8B
 - c. 16B
 - d. Ninguno de los anteriores
-

29. Un sistema tiene una cache asociativa por conjuntos de 2-vías con 16 conjuntos y líneas de 64B. ¿A qué conjunto le corresponde el byte con dirección 0xdeadbeef?

- a. 7
 - b. 11
 - c. 13
 - d. 14
-

30. ¿Cuál es el tamaño de la etiqueta de caché en un ordenador capaz de direccionar por bytes 1 MB de memoria principal y 32 KB de memoria caché y correspondencia asociativa por conjuntos con 32 bytes por línea y 16 líneas por conjunto?

- a. 6 bits
 - b. 7 bits
 - c. 8 bits
 - d. 9 bits
-

Nombre:	
DNI:	Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. En contraposición a un ejecutable Linux ELF, un fichero objeto (obtenido con gcc -c)
 - a. no contiene tablas de símbolos, que sólo se añaden al ejecutable tras enlazar
 - b. ubica el código (y los datos) a partir de la posición 0x0, las direcciones definitivas sólo se calculan tras enlazar
 - c. contiene tablas de símbolos, pero sólo para los símbolos locales; p.ej., sólo al enlazar un programa que llame a printf se añadirá una entrada a la tabla con la dirección de printf
 - d. no reserva espacio para las direcciones de objetos desconocidos; p.ej., sólo al enlazar un CALL a printf se insertan los 4B de la dirección de printf entre el codop de CALL y el de la siguiente instrucción máquina

2. Tras ejecutar las tres instrucciones que se muestran desensambladas a continuación, el registro EAX toma el valor

08048074 <_start>:
8048074: be 74 80 04 08 mov \$_start, %esi
8048079: 46 inc %esi
804807a: 8b 06 mov (%esi), %eax

 - a. 0x08048074
 - b. 0x08048075
 - c. 0x08048079
 - d. 0x0804807a

6. Para desplazar %eax a la derecha un número variable de posiciones ≤ 32 , indicado en %ebx, se puede hacer

- a. sar %ebx, %eax
- b. sar %bl, %eax
- c. **mov %ebx, %ecx**
sar %cl, %eax
- d. No se puede, sar sólo admite un número fijo de posiciones (debe ser un valor inmediato)

7. Indicar cuál es la dirección de salto (en qué dirección empieza la subrutina <main>) para esta instrucción call

0804854e: e8 3d 06 00 00 call <main>
08048553: 50 pushl %eax

- a. **08048b90**
- b. 08048b8b
- c. 450a854e
- d. No se puede deducir de la información proporcionada, faltan datos

8. El ajuste de marco de pila que gcc (Linux/IA-32) prepara para todas las funciones consiste en las instrucciones

- a. movl %ebp, %esp
pushl %ebp
- b. movl %esp, %ebp
pushl %esp
- c. pushl %ebp
movl %esp, %ebp
- d. pushl %esp
movl %ebp, %esp

9. Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/IA-32 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como

- a. var (el nombre de la variable representa su posición de memoria)
- b. **-k(%ebp)**, siendo k un número constante positivo relativamente pequeño
- c. **k(%ebp)**, siendo k un número constante positivo relativamente pequeño

d. **k(%esp)**, siendo k un número constante positivo relativamente pequeño

10. En x86_64 se pueden referenciar los registros

- a. **%rax, %eax, %ax, %ah, %al**
- b. %rsi, %esi, %si, %sih, %sil
- c. %r8, %r8d, %r8w, %r8l
- d. %r12q, %r12d, %r12w, %r12l

11. Comparando las convenciones de llamada de gcc Linux IA-32 con x86_64 respecto a registros

- a. En IA-32 %ebx es salva-invocante, pero en x86_64 %rbx es salva-invocado
- b. **En IA-32 %ecx es salva-invocante, y en x86_64 %rcx es salva-invocante también**
- c. En IA-32 %esi es salva-invocado, y en x86_64 %rsi es salva-invocado también
- d. En IA-32 %ebp es especial (marco de pila), y en x86_64 %rbp también

12. Si declaramos **int val[5]={1,5,2,1,3};** entonces

- a. **&val[2]** es de tipo int* y vale lo mismo que **val+8**
- b. **val+4** es de tipo int* y se cumple que ***(val+4)==5**
- c. **val+1** es de tipo int y vale 2
- d. **val[5]** es de tipo int y vale 3

13. Al traducir la sentencia C **r->i = val;** gcc genera el código ASM **movl %edx, 12(%eax)**. Se deduce que

- a. r es un puntero que apunta a la posición de memoria 12
- b. **el desplazamiento de i en *r es 12**
- c. i es un entero que vale 12
- d. val es un entero que vale 12

14. Una función C llamada **get_el()**

genera el siguiente código ensamblador.
Se puede adivinar que

```
movl 8(%ebp), %eax
leal (%eax,%eax,4), %eax
addl 12(%ebp), %eax
movl var(,%eax,4), %eax
```



Tienda 4

Tienda 3

Tienda 2

Tienda 1

- a. var es un array multi-nivel (punteros a enteros) de cuatro filas
 - b. var es un array multi-nivel pero no se pueden adivinar las dimensiones
 - c. var es un array bidimensional de enteros, no se pueden adivinar dimensiones
 - d. var es un array bidimensional de enteros, con cinco columnas**
-

15. En la práctica “suma” se pide sumar una lista de 32 enteros **con** signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando *overflow*. ¿Cuál es el menor valor positivo que repetido en toda la lista causaría *overflow* con 32bits?

- a. 0x0400 0000**
 - b. 0x0800 0000
 - c. 0x4000 0000
 - d. 0x8000 0000
-

16. En la práctica “suma” se pide sumar una lista de 32 enteros **con** signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando *overflow*. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista causaría *overflow* con 32bits?

- a. 0xffff ffff
 - b. 0xfc00 0000
 - c. 0xfbff ffff**
 - d. 0xf000 0000
-

17. La práctica “*parity*” debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. La siguiente función contiene un único error (en realidad se han editado 2 líneas de código sobre la versión correcta) pero produce resultado **correcto** cuando se usa como test el array

```
int parity4(unsigned* array,
            int len) {
    int i;
    unsigned x;
    int val=0, result=0;

    for (i=0; i<len; i++) {
        x = array[i];
        result ^= x;
    }
}
```

```
asm("\n"
"ini:                                \n\t"
"    xor %[x], %[v]                  \n\t"
"    shr %[x]                       \n\t"
"    jnz ini                         \n\t"
"        : [v]"+"r" "(val)"          \n\t"
"        : [x] "r" "(x)"             \n\t"
"    );                               \n"
"    result += val & 0x1;           \n"
"}                                \n"
"return result;                     \n"
}
```

- a. array={0, 1, 2, 3}
 - b. array={1, 2, 4, 8}
 - c. array={1, 16, 256, 1024}
 - d. array={5, 4, 3, 2}**
-

18. En la práctica de la cache, el código de “line.cc” incluye la sentencia

```
for (unsigned long long line=1;
line<=LINE; line<<=1) { ... }
```

¿Qué objetivo tiene la expresión line<<=1?

- a. salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
 - b. duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior**
 - c. volver al principio del vector cuando el índice excede la longitud del vector
 - d. sacar un uno (1) por el stream line
-

19. En la práctica de la cache, el código de “size.cc” accede al vector saltando de 64 en 64. ¿Por qué?

- a. Porque cada elemento del vector ocupa 64 bytes
 - b. Para recorrer el vector más rápidamente
 - c. Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB
 - d. Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal**
-

20. En el programa “size.cc” de la práctica de la cache, si el primer escalón pasa de tiempo=2 para un tamaño de vector

menor que 32KB a tiempo=8 para un tamaño mayor que 32KB, podemos asegurar que:

- a. La cache L1 es cuatro veces más rápida que la cache L2.
 - b. La cache L1 es seis veces más rápida que la cache L2.
 - c. La cache L1 es al menos cuatro veces más rápida que la cache L2.
 - d. La cache L1 es como mucho cuatro veces más rápida que la cache L2.
-

Nombre:
DNI:
Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1B), media palabra (2B) y palabra (4B), si se almacena en la posición 0xBABC una palabra de valor -2, ¿qué se obtendría al consultar la media palabra en la posición 0xBABE?
 - a. 0
 - b. 1
 - c. -1
 - d. -2

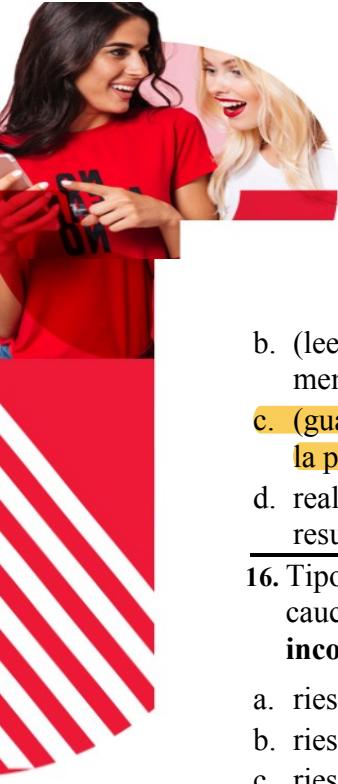
 2. Según la clasificación m/n, las máquinas de acumulador son de tipo
 - a. 0/0
 - b. 1/1
 - c. 1/2
 - d. 2/2 ó 2/3

 3. Según la clasificación m/n, las máquinas con arquitectura R/R son de tipo
 - a. 0/0
 - b. 0/x con x=2,3
 - c. x/0 con x=2,3
 - d. x/x con x=2,3

 4. El bus del sistema es
 - a. el que conecta las distintas partes del sistema: UC, ALU, E/S, M
- b. en un sistema con bus único, todo el bus salvo la parte relacionada con E/S (SATA, GPU, USB, Ethernet, etc)
- c. en un sistema con buses separados, el que conecta el sistema E/S con el resto
- d. el que conecta CPU-M, ya sea un sistema con bus único o con múltiples buses
-
5. Un sistema con direcciones de 8bits utiliza una puerta NAND conectada a las líneas A7...A5 para atacar la entrada CS# (activa baja) de un módulo de memoria. En el mapa de memoria las siguientes posiciones corresponderán a dicho módulo
 - a. 0x00 a 0x0f y 0x80 a 0x8f
 - b. 0x00 a 0x1f
 - c. 0x70 a 0x7f y 0xf0 a 0xff
 - d. 0xe0 a 0xff

 6. La ecuación básica de rendimiento calcula
 - a. cuánto tiempo tarda en ejecutarse un programa concreto conociendo su número de instrucciones y el número de etapas (promedio) y la frecuencia del procesador
 - b. cómo de mejor es un procesador frente a otro, conociendo las prestaciones de las respectivas UC, ALU, E/S y M
 - c. el promedio de las ganancias obtenidas con una serie de programas de punto entero

- d. la media geométrica de los cocientes entre los tiempos de ejecución de una serie de programas predeterminados
-
7. Por x86-64 se entiende la misma arquitectura de repertorio (ISA) que
- x86
 - IA-32
 - IA-64
 - AMD64**
-
8. ¿Qué combinación de *flags* aritmético-lógicos corresponde al código de condición b (*below*)?
- CF**
 - OF
 - CF xor OF
 - OF xor SF
-
9. En los modos de direccionamiento del tipo *Desplazamiento(Base,Indice,Factor Escala)*, puede usarse como
- desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)
 - base, cualquiera de los 8 registros enteros salvo %esp
 - índice, también cualquiera salvo %ebp
 - factor de escala, cualquier constante de 1, 2, 4 u 8 bytes
-
10. La diferencia entre las instrucciones **test** y **cmp** consiste en que
- test realiza una operación *and* lógico, mientras que cmp realiza una resta
 - test modifica sólo los *flags* lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
 - ambas respuestas son correctas**
 - ambas respuestas son incorrectas
-
11. Para traducir una construcción **if-then-else** de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente
- un salto condicional, según la condición expresada en el código C
 - un salto condicional, según la condición opuesta a la del código C, y otro salto incondicional
 - dos saltos condicionales (uno para la parte if y otro para la parte else)
 - dos saltos condicionales y dos saltos incondicionales
-
12. Para traducir una construcción **do-while** de lenguaje C a lenguaje ensamblador, gcc utiliza generalmente
- un salto condicional hacia adelante, según la misma condición que en lenguaje C
 - un salto condicional hacia atrás, según la misma condición que en lenguaje C
 - un salto condicional hacia adelante, según la condición opuesta a la de lenguaje C
 - un salto condicional hacia atrás, según la condición opuesta a la de lenguaje C
-
13. Alguna de las siguientes señales **no** sirve de entrada a la unidad de control. ¿Cuál?
- señal de reloj (CLK)
 - estado de la unidad de proceso (*flags* Z, S, C, O...)
 - instrucción actual (bits del registro IR)
 - contador de programa (bits del registro PC)**
-
14. Alguna de las siguientes señales **no** es salida de la unidad de control. ¿Cuál?
- señales de carga, habilitación y/o desplazamiento de registros (*Load, Enable, ShiftL, ShiftR*)
 - códigos de selección en multiplexores, decodificadores, ALU, etc (00, 01, 10, 11...)
 - señales de lectura y escritura en memoria (RD, WR)
 - dirección de la siguiente microinstrucción (bits del campo DIR o Memoria B de Wilkes)**
-
15. Alguna de las siguientes **no** es una operación básica de la unidad de control
- transferir un registro a otro



Clases de inglés online particulares y con profesores nativos. Enfocados en la conversación y especialistas en Business English.

Tu academia de inglés

talkingmethod.com

¿Quieres aprobar inglés de una vez?

Prepárate con las mejores clases online



Talking Method

- b. (leer o escribir) un registro (de / a) memoria
- c. (guardar o recuperar) un registro (en / de) la pila
- d. realizar operación ALU y guardar resultado en registro
-
16. Tipos de riesgos que hemos estudiado en cauces segmentados (señalar la opción **incorrecta**)
- a. riesgos estructurales
- b. riesgos de (dependencia de) datos
- c. riesgos de control
- d. riesgos de transferencia
-
17. ¿Cuál de los siguientes modos de direccionamiento es **menos** preferible para un procesador con segmentación de cauce?
- a. registro
- b. directo
- c. indirecto a través de registro
- d. indexado (o relativo a base, o base+índice)
-
18. La ganancia en velocidad ideal de un cauce de K etapas de igual duración T ejecutando un programa de N instrucciones es
- a. $S = KN/(K-N+1)$
- b. $S = NKT/(N-K+1)T$
- c. $S = KN/(K+N-1)$
- d. $S = NT/(N+K-1)T$
-
19. Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?
- a. el retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)
- b. el retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple
- c. el retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca
-
- d. el retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de salto
-
20. Respecto a las técnicas de direccionamiento por selección lineal, decodificación centralizada y distribuida
- a. todas ellas impiden que haya cortocircuito en el bus de datos
- b. todas ellas impiden que haya cortocircuito en el bus de direcciones
- c. la selección lineal permitiría escribir un mismo dato a varios puertos E/S
- d. usando decodificación centralizada es más fácil realizar expansiones al sistema de E/S
-
21. Ventajas de la E/S independiente (separada, aislada) (señalar la opción **incorrecta**)
- a. diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
- b. protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
- c. mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
- d. decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
-
22. Parecidos y diferencias entre los métodos de E/S (señalar la opción **incorrecta**)
- a. la consulta del estado del dispositivo por parte de la CPU se suele/puede hacer con E/S programada y con E/S por IRQ
- b. se suele avisar a la CPU (con una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ como en E/S por DMA
- c. sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo
- d. sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S
-

23. Tipos de interrupción que suelen contemplar las CPUs comerciales actuales (señalar la opción **incorrecta**)

- a. internas (excepciones o *traps*): generadas internamente por la CPU para indicar una condición que requiere atención (división por cero, *codop* inválido, etc)
- b. externas (IRQs *hardware*): generadas por un dispositivo externo a la CPU, activan la línea INTR# (o equivalente)
- c. software: generadas al ejecutar la instrucción INT (o equivalente)
- d. **firmware (faults)**: generadas por el microcódigo de la CPU (*segmentation fault, page fault, etc*)

24. Técnicas que se pueden usar para determinar la causa de una interrupción (señalar la opción **incorrecta**)

- a. múltiples líneas de interrupción INT1#, INT2#...
- b. **línea de reconocimiento INTA#**
- c. consulta de estado, o *polling*
- d. interrupciones vectorizadas

25. Respecto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR)

- a. no es necesario salvar ninguno más, si el contador de programa y los *flags* de estado ya los salva la propia CPU como parte del mecanismo de interrupción
- b. **se deben guardar los registros que se modifiquen en la propia ISR**. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar
- c. se deben guardar los registros salvado-
invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA-32), los registros salvado-
invocante ya los guarda el programa
interrumpido
- d. se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción

26. En EC podemos usar la palabra *directo* para referirnos a... (señalar la opción **incorrecta**)

- a. **dispositivo de almacenamiento secuencial directo (DASD)**
- b. cache con correspondencia directa
- c. acceso directo a memoria
- d. modo de direccionamiento directo

27. Variación de los parámetros de los distintos niveles en una jerarquía de memoria (señalar la opción **incorrecta**)

- a. **unidad de transferencia: $x_i \geq x_{i+1}$**
- b. ancho de banda: $b_i \geq b_{i+1}$
- c. tamaño del nivel: $s_i \leq s_{i+1}$
- d. tiempo de acceso: $t_i \leq t_{i+1}$

28. Las técnicas *write-through* y *write-back* están relacionadas con

- a. **coherencia de cache**
- b. métodos de E/S
- c. arbitraje de buses
- d. etapas de la unidad de control

29. Para diseñar una memoria con ancho de palabra $k \cdot m$ (y mismo nº palabras que los módulos) a partir de módulos con ancho de palabra m , se utilizan k módulos

- a. repartiendo las líneas de datos entre los k módulos: el primero se conecta a $D_0 \dots D_{k-1}$, el segundo a $D_k \dots D_{2k-1}$, etc
- b. repartiendo las líneas de dirección: el 1º se conecta a $A_0 \dots A_{k-1}$, el 2º a $A_k \dots A_{2k-1}$, etc
- c. **repartiendo líneas datos: el 1º se conecta a $D_0 \dots D_{m-1}$, el 2º a $D_m \dots D_{2m-1}$, etc**
- d. repartiendo líneas dirección: el 1º a $A_0 \dots A_{m-1}$, el 2º a $A_m \dots A_{2m-1}$, etc

30. En un sistema con direcciones de 32bits, memoria de bytes, cache de 1MB asociativa por conjuntos de 4 vías y líneas de 64B, el campo etiqueta en el formato de dirección cache es de

- a. 16bits
- b. **14bits**
- c. 12bits
- d. 10bits

Nombre:

DNI:

Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. La dirección efectiva del primer parámetro de llamada a una función suele calcularse desde el código de la función como:
 - a. EBP+8
 - b. EBP-8
 - c. EBP+4
 - d. EBP-4

2. El comienzo de un procedimiento que siga la convención cdecl es:
 - a. mov %ebp,%esp; push %ebp
 - b. mov %esp,%ebp; push %ebp
 - c. push %ebp; mov %ebp,%esp
 - d. **push %ebp; mov %esp,%ebp**

3. Considere una función C declarada así:

```
void fun4arg (int a,int b,int c,int d);
```

Suponiendo que fun4arg se ha compilado para una máquina x86 IA-32 con enteros de 4 bytes, ¿cuál sería la dirección del argumento b relativa a %ebp, en el marco de pila de fun4arg?

 - a. %ebp + 8
 - b. %ebp + 12**
 - c. %ebp + 16
 - d. %ebp + 20

4. ¿Cuál de las siguientes afirmaciones sobre las caches es ***FALSA***?
 - a. **Casi ningún procesador actual tiene memoria cache L2**

- b. Las direcciones a las que accede un programa no son completamente aleatorias, sino que se rigen por ciertos patrones de localidad
- c. Un procesador actual tiene varias cachés de nivel 1
- d. La caché de nivel 3 no contiene toda la memoria que maneja el programa

5. En un sistema Linux x86-64, ¿cuál de las siguientes variables ocupa más bytes en memoria?
 - a. char a[7]
 - b. short b[3]
 - c. int *c**
 - d. float d

6. En la práctica "suma" se pide sumar una lista de 32 enteros SIN signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando acarreos. ¿Cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32bits (sin signo)?
 - a. 0xfc00 0000
 - b. 0xfbff ffff
 - c. 0x0800 0000**
 - d. 0x07ff ffff

7. En la práctica "suma" se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando

desbordamiento. ¿Cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32bits (en complemento a 2)?

- a. 0xfc00 0000
- b. 0xfbff ffff**
- c. 0xf800 0000
- d. 0xf800 0001

8. ¿Qué valor contendrá edx tras ejecutar las siguientes instrucciones?

```
xor %eax, %eax  
sub $1, %eax  
cltd  
idiv %eax
```

- a. 0**
- b. 1
- c. -1
- d. No puede saberse con los datos del enunciado

9. La práctica "popcount" debía calcular la suma de bits de los elementos de un array. Un estudiante entrega lo siguiente:

```
int popcount4(unsigned* array,  
              int len) {  
    int i, j, res = 0;  
    for(i = 0; i < len; ++i) {  
        unsigned x = array[i];  
        int n = 0;  
        do {  
            n += x & 0x01010101L;  
            x >>= 1;  
        } while(x);  
        for(j = 16; j == 1; j /= 2){  
            n ^= (n >>= j);  
        }  
        res += n & 0xff;  
    }  
    return res;  
}
```

Esta función popcount4:

- a. produce el resultado correcto**
- b. fallaría con **array={0,1,2,3}**
- c. fallaría con **array={1,2,4,8}**
- d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

10. La práctica "paridad" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned * array,  
            int len) {  
    int i, result = 0;  
    unsigned x;  
    for (i=0; i<len; i++) {  
        x = array[i];  
        asm("mov %[x], %%edx\n\t"  
             "shr $16, %%edx\n\t"  
             "shr $8, %%edx\n\t"  
             "xor %%edx,%%edx\n\t"  
             "setp %%dl\n\t"  
             "movzx %%dl, [%x]\n\t"  
             : [x] "+r" (x)  
             :  
             : "edx"  
        );  
        result += x;  
    }  
    return result;  
}
```

Esta función parity6:

- a. produce el resultado correcto
- b. fallaría con **array={0,1,2,3}****
- c. fallaría con **array={1,2,4,8}**
- d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

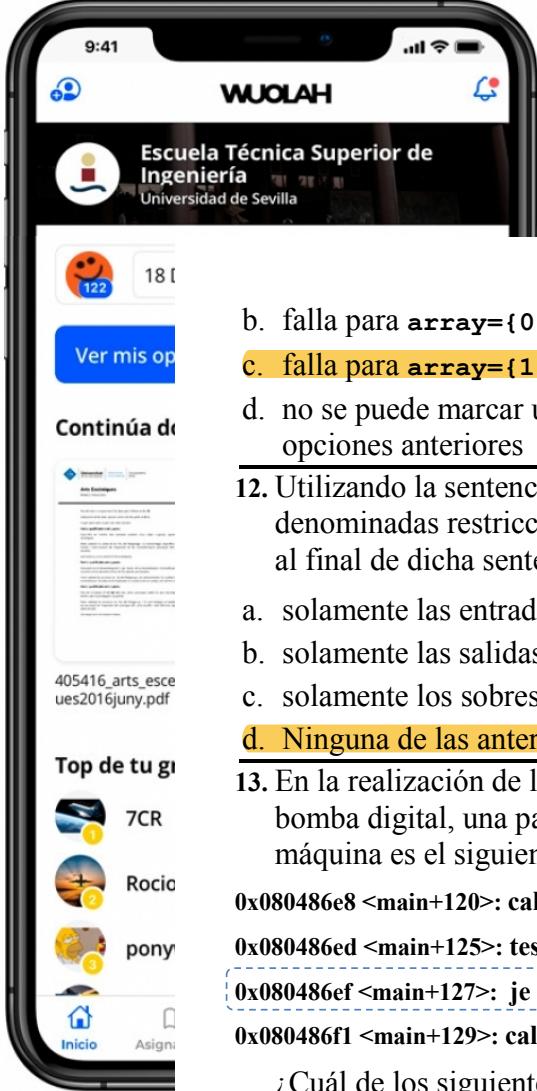
11. En la práctica "paridad" se pide calcular la suma de paridades de una lista de enteros sin signo. Suponer que un estudiante entrega la siguiente versión:

```
int paridad5(unsigned* array,  
              int len) {  
    int i, k, result = 0;  
    unsigned x;  
    for (i = 0; i < len; i++) {  
        x = array[i];  
        for (k = 16; k == 1; k /= 2)  
            x ^= x >> k;  
        result += (x & 0x01);  
    }  
    return result;  
}
```

Esta función:

- a. es correcta





Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- b. falla para `array={0,1,2,3}`
- c. falla para `array={1,2,3,4}`**
- d. no se puede marcar una y sólo una de las opciones anteriores

12. Utilizando la sentencia asm, las denominadas restricciones que se indican al final de dicha sentencia, involucran a:

- a. solamente las entradas
- b. solamente las salidas
- c. solamente los sobrescritos
- d. Ninguna de las anteriores es cierta**

13. En la realización de la práctica de la bomba digital, una parte del código máquina es el siguiente:

```
0x080486e8 <main+120>: call 0x8048524 <strcmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6<main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
```

¿Cuál de los siguientes comandos cambiaría el salto condicional por un salto incondicional?

- a. set \$0x080486ef=0xeb
- b. set *(char*)0x080486ef=0xeb**
- c. set *(char*)0x080486f6=jmp
- d. set %0x080486ef=0xeb

14. En una bomba como las estudiadas en prácticas, del tipo...

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756<main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

- a. el entero 0x804a044
- b. el entero almacenado a partir de la posición de memoria 0x804a044**
- c. el string almacenado a partir de la posición de memoria 0x24(%esp)
- d. ninguna de las anteriores

15. En una bomba como las estudiadas en prácticas, del tipo...

```
0x080486e8 <main+120>: call 0x8048524 <strcmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
0x080486f6 <main+134>: ...
```

la contraseña es...

- a. el valor que tenga %eax
- b. el string almacenado a partir de donde apunta %eax
- c. el entero almacenado a partir de donde apunta %eax
- d. ninguna de las anteriores**

16. El servidor de SWAD tiene dos procesadores Xeon E5540 con 4 núcleos cada uno. Cada procesador tiene 4 caches L1 de instrucciones de 32 KB, 4 caches L1 de datos de 32 KB, 4 caches unificadas L2 de 256 KB y una cache unificada L3 de 8MB. Suponga que un proceso swad, que se ejecuta en un núcleo, tiene que ordenar un vector de estudiantes accediendo repetidamente a sus elementos. Cada elemento es una estructura de datos de un estudiante y tiene un tamaño de 4KB. Si representamos en una gráfica las prestaciones en función del número de estudiantes a ordenar, ¿para qué límites teóricos en el número de estudiantes se observarán saltos en las prestaciones debidos a accesos a la jerarquía de memoria?

- a. 4 / 32 / 512 estudiantes
- b. 8 / 64 / 2048 estudiantes**
- c. 16 / 32 / 64 estudiantes
- d. 32 / 256 / 8192 estudiantes

17. En la práctica de la cache, el código de line.cc incluye la sentencia

```
for (unsigned line=1;line<=MAXLINE;
     line<<=1) { ... }
```

¿Qué objetivo tiene la expresión line<<=1?

WUOLAH

- a. Salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
- b. Duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior
- c. Volver al principio del vector cuando el índice exceda la longitud del vector
- d. Sacar un uno (1) por el stream line

18. Sea un computador de 32 bits con una memoria caché L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:

```
int v[262144];
for (i = 0; i < 262144; i += 2)
    v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la primera ejecución del bucle anterior?

- a. 0 (ningún fallo)
- b. 1/2 (mitad aciertos, mitad fallos)
- c. 1/8 (un fallo por cada 8 accesos)
- d. 1 (todo son fallos)

19. Abajo se ofrece el listado de una función para multiplicar matrices $C = A \times B$.

```
void mult_matr(    float A[N][N],
                  float B[N][N], float C[N][N]) {
    /* Se asume valor inicial C = {0,0...} */
    int i,j,k;
    for (i=0; i<N; i++)
        for (j=0; j<N; j++)
            for (k=0; k<N; k++)
                C[i][j] += A[i][k] * B[k][j];
}
```

Suponer que:

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.
- El tamaño de los tipos de datos es como en IA32.
- El compilador optimiza el acceso a $C[i][j]$ en un registro.

Aproximadamente, ¿qué tasa de fallos se podría esperar de esta función para valores grandes de N?

- a. 1/16
- b. 1/8
- c. 1/4
- d. 1/2

20. Con los mismos supuestos, imaginar que se modifica la última sentencia (el cuerpo anidado) por esta otra

```
C[i][j] += A[i][k] * B[j][k];
```

de manera que se calcule $C = A \times B'$ (A por traspuesta de B). Aproximadamente, ¿qué tasa de fallos se podría esperar de esta nueva función para valores grandes de N?

- a. 1/16
- b. 1/8
- c. 1/4
- d. 1/2

Nombre:
DNI:
Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. Si almacenamos según el criterio little-endian la palabra de 64 bits 0xFACEB00C a partir de la dirección 0xCAFEBAE, el byte 0xCE quedará almacenado en la dirección:
 a. 0xCAFEBA1
 b. **0xCAFEBA0**
 c. 0xCAFEBABF
 d. 0xCAFEBAE

2. ¿Qué novedad se desarrolló en la tercera generación de computadores?
 a. **Los circuitos integrados**
 b. Los primeros lenguajes de programación de alto nivel
 c. Los microprocesadores CISC
 d. Los microprocesadores RISC

3. En X86-64, el registro contador de programa se denomina:
 a. **RIP**
 b. EIP
 c. IP
 d. R15

4. La primera letra (l) de la instrucción lea:
 a. indica que la instrucción usa ordenación de bytes little-endian
 b. indica que la instrucción trabaja con un operando destino de 32 bits (long word)
 c. indica que la instrucción afecta a los 16 bits menos significativos del operando destino (low word)
 d. **forma parte del nemotécnico de la instrucción**

5. ¿Cuál es la diferencia entre las instrucciones mov y lea?
 a. lea accede a la posición indicada, mientras que mov no lo hace
 b. **mov accede a la posición indicada, mientras que lea no lo hace**
 c. lea puede usarse para copiar un registro a otro, mientras que mov no
 d. mov puede usarse para copiar un registro a otro, mientras que lea no

6. La instrucción not:
 a. realiza el complemento a dos
 b. **realiza el complemento a uno (cambiar unos por ceros y ceros por unos)**
 c. realiza la operación no-or (or negada)
 d. realiza un salto condicional si negativo

7. La instrucción JGE / JNL provoca un salto si...
 a. SF = 1
 b. CF = 1
 c. SF = 0
 d. **OF = SF**

8. Un overflow nunca puede ocurrir cuando:
 a. se suman dos números positivos
 b. se suman dos números negativos
 c. **se suma un número positivo a un número negativo**
 d. se resta un número positivo de un número negativo

9. Despues de ejecutar una instrucción de suma sobre dos números con signo de la que

sabemos que no provocará overflow (los dos números son pequeños en valor absoluto), queremos comprobar si el resultado de la suma es menor que 0. ¿Qué flag necesita comprobar la instrucción de salto condicional equivalente a “if (resultado<0) then goto label”?

- a. CF
- b. OF
- c. SF
- d. ZF

10. La instrucción cmovb %edx, %eax

- a. copia en %eax el contenido de %edx si el indicador de acarreo es 1
- b. copia el byte bajo de %edx en el byte bajo de %eax
- c. copia en %eax el byte de memoria apuntado por la dirección contenida en %edx
- d. copia en %eax el contenido de %edx si %eax es menor que %edx

11. ¿Cuál de las siguientes afirmaciones sobre la instrucción leave es cierta?

- a. Se ejecuta justo después de retornar de un procedimiento
- b. Equivale a pop %ebp seguida de mov %ebp,%esp
- c. Equivale a mov %esp,%ebp seguida de pop %ebp
- d. No es obligatorio usarla. En su lugar puede realizarse una secuencia explícita de operaciones mov y pop

12. Para crear espacio en la pila para variables locales sin inicializar suele realizarse la siguiente operación:

- a. Restar una cantidad positiva a EBP
- b. Sumar una cantidad positiva a EBP
- c. Restar una cantidad positiva a ESP
- d. Sumar una cantidad positiva a ESP

13. ¿Cuál de los siguientes lenguajes no permite el paso de parámetros por referencia?

- a. Pascal
- b. C
- c. C++
- d. FORTRAN

14. En la secuencia de programa siguiente:

```
804854e:e8 3d 06 00 00 call 8048b90 <main>
8048553:50      pushl %eax
```

¿cuál es el valor que introduce en la pila la instrucción call?

- a. 804854e
- b. 804854f
- c. 8048b90
- d. 8048553

15. En el fragmento de código

```
804854e:e8 3d 06 00 00 call 8048b90 <main>
8048553:50      pushl %eax
```

la instrucción call suma al contador de programa la cantidad:

- a. 0000063d
- b. 08048553
- c. 0804854e
- d. 50

16. Es responsabilidad del procedimiento llamado salvaguardar los registros:

- a. %ebx, %esi, %edi
- b. %eax, %edx, %ecx
- c. %eax, %ebx, %ecx, %edx
- d. %esi, %edi

17. Al llamar a una función de 2 argumentos foo(arg1, arg2) , ¿cuál es el orden correcto en el que se ejecutan las instrucciones? (suponiendo convención de llamada x86 cdecl, y que foo requiere ajustar marco de pila, esto es, salvar %ebp)

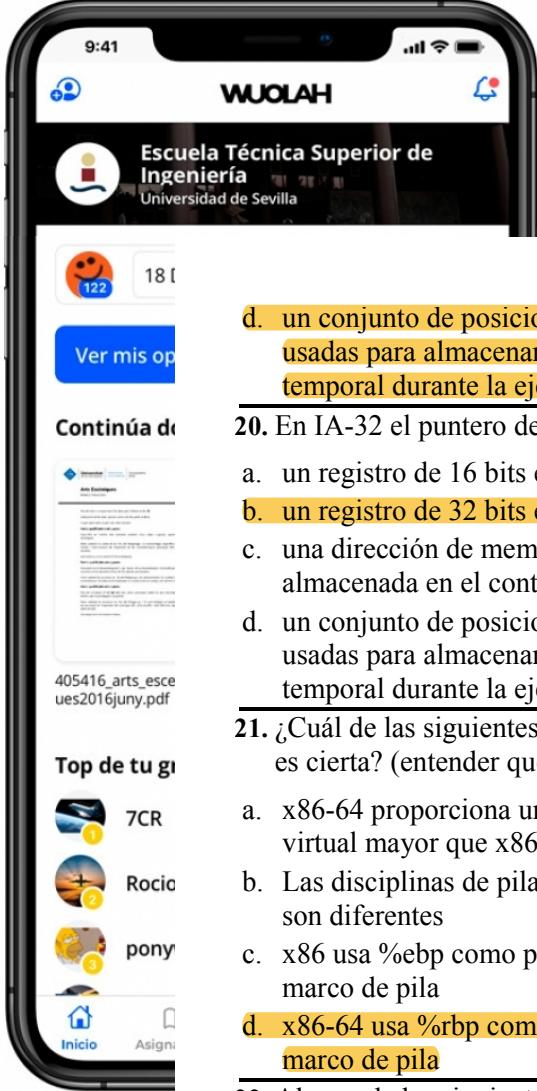
- a. push arg1, push arg2, call foo, push %ebp
- b. push arg1, push arg2, push %ebp , call foo
- c. push arg2, push arg1, call foo, push %ebp
- d. push arg2, push arg1, push %ebp, call foo

18. Cuando se ejecuta la instrucción ret al final de una subrutina:

- a. la dirección de comienzo de la pila se transfiere al puntero de pila
- b. la dirección de memoria de la instrucción ret se transfiere al contador de programa
- c. la dirección almacenada en la cima de la pila se transfiere al contador de programa
- d. la dirección almacenada en la cima de la pila se transfiere al puntero de pila

19. En IA-32 la pila es:

- a. un registro de 16 bits en el microprocesador
- b. un registro de 32 bits en el microprocesador
- c. una dirección de memoria de 32 bits almacenada en el contador de programa



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- d. un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa

20. En IA-32 el puntero de pila es:

- a. un registro de 16 bits en el microprocesador
b. un registro de 32 bits en el microprocesador
c. una dirección de memoria de 32 bits almacenada en el contador de programa
d. un conjunto de posiciones de memoria usadas para almacenar información temporal durante la ejecución del programa

21. ¿Cuál de las siguientes afirmaciones *NO* es cierta? (entender que x86=IA-32)

- a. x86-64 proporciona un espacio de memoria virtual mayor que x86
b. Las disciplinas de pila para x86 y x86-64 son diferentes
c. x86 usa %ebp como puntero base para el marco de pila
d. x86-64 usa %rbp como puntero base para el marco de pila

22. Alguna de las siguientes *NO* es una operación básica de la unidad de control:

- a. Transferir un registro a otro
b. (Leer / escribir) un registro (de / a) memoria
c. (Guardar / recuperar) registro (en/de) la pila
d. Realizar una operación de la ALU y guardar el resultado en un registro

23. Un computador tiene una memoria de control de 16000 palabras de 250 bits, de las que 447 son diferentes. ¿Cuántos bits ahorraremos usando nanoprogramación en lugar de microprogramación?

- a. 3744250
b. 259206
c. 287935
d. Ninguno de los resultados anteriores es exacto

24. Un sistema no segmentado tarda 20 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se aproxima a:

- a. 5
b. 4
c. 0,25

- d. 20

25. En la técnica de salto retardado:

- a. el compilador puede reorganizar el código para llenar los huecos de retardo con instrucciones útiles
b. el compilador no puede insertar operaciones NOP en los huecos de retardo
c. el salto se realiza varios ciclos antes de la instrucción de salto
d. las instrucciones en los huecos de retardo se ejecutan unas veces y otras no

26. Al método de interacción con los periféricos, en los que el procesador vigila periódicamente el estado de los dispositivos mediante una encuesta activa se le denomina:

- a. daisy-chain
b. interrupción
c. polling
d. DMA

27. ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

- a. 1
b. 2
c. 3
d. 4

28. Se dispone de un procesador con una frecuencia de reloj de 1 GHz. Se le conecta un dispositivo que genera 100.000 interrupciones por segundo. La rutina de servicio de interrupción ejecuta 500 instrucciones. El número medio de ciclos por instrucción es 2. ¿Qué porcentaje del tiempo dedica el procesador al dispositivo?

- a. 1%
b. 10%
c. 50%
d. 90%

29. ¿Cuál de las siguientes afirmaciones acerca de la memoria es *FALSA*?

- a. La memoria dinámica usa señales de control RAS# y CAS#
b. Las celdas de memoria dinámica están constituidas por un transistor y un condensador

c. Las celdas de memoria estática tienen que ser constantemente refrescadas

d. La memoria estática se emplea en las cachés L1 y L2

30. ¿Cuál de las siguientes afirmaciones acerca de la jerarquía de memoria es ***FALSA***?

a. Acceder a los discos es órdenes de magnitud más lento que acceder a la RAM

b. Una memoria principal constituida por la tecnología más rápida es órdenes de magnitud más cara que la DRAM

c. La velocidad de acceso a la memoria principal ha crecido proporcionalmente a la velocidad del procesador

d. Un computador puede tener una pequeña cantidad de memoria rápida además de una gran cantidad de memoria más lenta

Nombre:	
DNI:	Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. El switch de gcc para que únicamente compile de lenguaje C a ensamblador, y no realice ningún paso adicional (ensamblar, enlazar, etc), es...
 - a. -c
 - b. **-S**
 - c. -o
 - d. -g

2. Los switches --32 y --64 para trabajar en 32bit/64bit corresponden a la herramienta...
 - a. gcc
 - b. **as**
 - c. ld
 - d. nm

3. El switch -l para indicar librerías ***NO*** funciona con la herramienta...
 - a. gcc
 - b. **as**
 - c. ld
 - d. no se puede marcar una y solo una de las anteriores

4. ¿Cuál de las siguientes no es una sección de un fichero ELF?
 - a. .text
 - b. **.static**
 - c. .data
 - d. .bss

5. ¿Cuál de los siguientes contenidos no está incluido en un fichero ELF ejecutable?
 - a. código máquina
 - b. variables globales
 - c. **pila del usuario**
 - d. tabla de símbolos

6. En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor que se indica a continuación, ¿en qué caso ambos programas producen el mismo resultado?
 - a. 0x1111 1111
 - b. 0x9999 9999
 - c. 0xAAAA AAAA
 - d. 0xFFFF FFFF

7. En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x0400 0000, ¿en qué se diferencian los resultados de ambos programas?
 - a. **no se diferencian**
 - b. en uno ocupa 32 bits, en otro 64 bits
 - c. en uno se interpreta como negativo, en otro como positivo
 - d. en uno los 32 bits superiores son 0xFFFF FFFF, en el otro no

8. En la práctica "media" se suma una lista de 32 enteros de 4 B con signo para producir una media y un resto usando la instrucción IDIV. ¿Cuál de las siguientes afirmaciones es falsa?

- a. IDIV produce el mismo cociente que el operador / en lenguaje C
- b. IDIV produce el mismo resto que el operador % en lenguaje C
- c. La media se redondea al entero más próximo
- d. El resto siempre tiene el mismo signo que la suma

9. En la práctica "media" un estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

bucle:

```
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %edx, %ebp
    jnc nocarry
    inc %edx

nocarry:
    inc %esi
    cmp %esi,%ecx
    jne bucle
```

Estando bien programado todo lo demás, este código

- a. produce siempre el resultado correcto
- b. fallaría con lista: .int 0,1,2,3
- c. fallaría con lista: .int -1,-2,-4,-8
- d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

10. Alguno de los siguientes no es un nombre de registro en una máquina IA-32 en modo 32 bits

- a. ebp
- b. ax
- c. dh
- d. sil

11. Alguno de los siguientes no es un nombre de registro en una máquina x86-64 en modo 64 bits

- a. r8d
- b. r12w
- c. sih
- d. spl

12. Para comprobar si el entero almacenado en EAX es cero (y posiblemente saltar a continuación usando JZ/JNZ), gcc genera el código

- a. cmp %eax, \$0
- b. test %eax
- c. cmp %eax
- d. test %eax, %eax

13. La práctica "paridad" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity3:

```
int parity3(unsigned* array,
            int len){
    int i,res=0,val;
    unsigned x;
    for (i=0; i<len; i++){
        x=array[i];
        val=0;
        do {
            val += x;
            x >>= 1;
        } while (x);
        val &= 0x1;
        res+=val;
    }
    return res;
}
```

Esta función parity3:

- a. produce siempre el resultado correcto
- b. fallaría con array={0,1,2,3}
- c. fallaría con array={1,2,4,8}
- d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

- **480** cartas resultado de mucho amor y birra
- **80 cartas especiales** para animar tus fiestas
- **A partir de 16 años**, de 3 a 15 jugadores
- Perfecto para largas noches de risas con amigos



14. Un estudiante entrega la siguiente versión de parity4:

```
int parity4(unsigned* array,
            int len) {
    int val,i,res=0;
    unsigned x;
    for (i=0; i<len; i++) {
        x=array[i];
        val=0;
        asm("\n"
"ini3:           \n\t"
"    xor  %[x],%[v] \n\t"
"    shr  %[x]          \n\t"
"    test %[x], %[x]\n\t"
"    jne   ini3      \n\t"
"    :[v]" "+r" (val)
"    :[x] "r" (x)
);
        val = val & 0x1;
        res+=val;
    }
    return res;
}
```

Esta función parity4:

- produce siempre el resultado correcto
- fallaría con array={0,1,2,3}
- fallaría con array={1,2,4,8}
- no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

15. La sentencia `asm()` del listado anterior tiene las siguientes restricciones

- ninguna
- arquitectura de 32 bits
- dos entradas y una salida**
- un registro y dos sobreescritos (clobber)

16. Un estudiante entrega la siguiente versión de parity5:

```
int parity5(unsigned* array,
            int len) {
    int i,j,res=0;
    unsigned x;
    for (i=0; i<len; i++) {
        x=array[i];
        for (j=sizeof(unsigned)*4;
```

```
j>0; j=j/2) {
    x^=x>>j;
}
x = x & 0x1;
res+=x;
}
return res;
}
```

Esta función parity5:

- produce siempre el resultado correcto
- fallaría con array={0,1,2,3}
- fallaría con array={1,2,4,8}
- no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

17. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned* array,
            int len) {
    int i,j,res=0;
    unsigned x;
    for (i=0; i<len; i++) {
        x=array[i];
        asm("\n"
"    mov  %[x],%%edx \n\t"
"    shr $16, %%edx \n\t"
"    xor %%edx,%[x] \n\t"
"    mov  %[x],%%edx \n\t"
"    mov %%dh, %%dl \n\t"
"    xor %%edx, %[x]\n\t"
"    setpo %%cl      \n\t"
"    movzx %%cl, %[x]"
"    :[x] "+r" (x)
"    :"
"    :edx", "ecx"
);
        res+=x;
    }
    return res;
}
```

Esta función parity6:

- produce siempre el resultado correcto
- fallaría con array={0,1,2,3}
- fallaría con array={1,2,4,8}

d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

18. La sentencia `asm()` del listado anterior tiene las siguientes restricciones

- a. ninguna
 - b. arquitectura de 32 bits
 - c. dos entradas y una salida
 - d. un registro y dos sobrescritos (clobber)**
-

19. En el programa "size" de la práctica de la cache, si el primer escalón pasa de tiempo = 1 para todos los tamaños de vector menores o iguales que 32 KB a tiempo = 3 para los tamaños 64 KB y 128 KB, podemos asegurar que:

- a. la cache L1 es al menos tres veces más rápida que la cache L2.**
 - b. la cache L1 es como mucho tres veces más rápida que la cache L2.
 - c. la cache L2 es al menos el doble de rápida que la memoria principal.
 - d. la cache L2 es como mucho el doble de rápida que la memoria principal.
-

20. El código del programa "size" de la práctica de la cache accede al vector saltando...

- a. de byte en byte.
 - b. de 64 en 64 bytes.**
 - c. de 1 KB en 1 KB.
 - d. de 64 KB en 64 KB.
-

Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. El conjunto de todos los atributos de un sistema que son visibles para el programador y son necesarios para programar en lenguaje máquina se denomina:
 - a. **arquitectura del computador**
 - b. conjunto de componentes físicos del computador
 - c. organización del computador
 - d. repertorio de instrucciones máquina

2. ¿Cuál de las siguientes afirmaciones es cierta?
 - a. la arquitectura Von Neumann de los computadores tradicionales consiste en tener almacenados los datos separados de las instrucciones en memorias distintas
 - b. **el registro de estado (flags) es un registro de propósito específico cuyo contenido puede ser visto directa o indirectamente por el usuario mediante el uso de ciertas instrucciones específicas**
 - c. la unidad de control necesita como entrada el registro contador de programa para saber cuál es la instrucción que debe ejecutar a continuación
 - d. el registro de direcciones de memoria es un registro de propósito general que puede contener tanto direcciones como datos

3. En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1 B), media palabra (2 B) y palabra (4 B), se almacenan a partir de la posición 0xCAFEBAB0 cuatro palabras con valores -1, -2, -3, -4. ¿Qué se obtendría al

consultar la media palabra de la posición 0xCAFEBABE?

- a. **-1**
- b. -4
- c. no se puede saber, faltan datos
- d. ninguna de las anteriores

4. Se pretende almacenar una palabra de 4 B en una memoria de bytes a partir de una dirección determinada. ¿Cuál de las siguientes es válida, si la palabra debe quedar alineada?
 - a. **0xFACEB00C**
 - b. 0xDEADBEEF
 - c. 0xCAFEBAE
 - d. 0xABADF00D

5. En una arquitectura de acumulador, la instrucción LOAD X:
 - a. transfiere el contenido del registro X a la memoria
 - b. suma M(X) al acumulador
 - c. transfiere el contenido del acumulador a la posición de memoria X
 - d. **transfiere el contenido de la posición de memoria X al acumulador**

6. Una instrucción máquina del tipo "Add M,R" podría formar parte del repertorio de
 - a. una máquina pila
 - b. una máquina de acumulador
 - c. una máquina con arquitectura R/R
 - d. **una máquina con arquitectura M/M**

7. ¿Cuál de los siguientes no es un modo de direccionamiento IA-32?
- Registro
 - Memoria
 - Cache**
 - Inmediato
-
8. Un bus se compone de:
- líneas de datos y líneas de dirección
 - líneas de alimentación
 - líneas de estado y líneas de control
 - líneas de control/estado, líneas de dirección y líneas de datos**
-
9. ¿Cuál de los siguientes no es un tipo de bus?
- Secuencial**
 - Paralelo
 - E/S
 - Sistema
-
10. Si en un bus de direcciones de 32 bits se decodifica parcialmente la dirección de un dispositivo de 32 posiciones usando 22 bits, ¿cuántas veces aparecerá repetido en el mapa de memoria?
- 10
 - 16
 - 32**
 - 1024
-
11. Para obtener una única velocidad comparativa final, el benchmark SPEC CPU combina las velocidades de ejecución de una serie de tests, respecto a un ordenador de referencia, usando la media...
- aritmética
 - geométrica**
 - armónica
 - ponderada
-
12. El primer computador electrónico basaba su funcionamiento en:
- tubos de vacío**
 - circuitos integrados LSI
 - amplificadores operacionales
 - núcleos de ferrita
-
13. En Linux IA-32, si gcc usa la instrucción leave se puede asegurar que en ese punto del programa
- correspondería emitir la secuencia de salida pop/ret, pero leave hace lo mismo y ocupa menos espacio
 - ya no hay registros salva-invocado que recuperar**
 - ya no hay variables locales que destruir
 - ya no se hacen llamadas anidadas y por tanto no hay parámetros que ocupen espacio en pila
-
14. Usando el repertorio IA-32, para intercambiar el valor de 2 variables (por ejemplo A: .int 1 y B: .int 2) se pueden usar...
- dos instrucciones mov
 - una instrucción mov y una instrucción lea
 - 3 mov, no menos (se le llama "intercambio circular")
 - 4 mov, no menos (debido a la arquitectura R/M)**
-
15. Respecto a registros base e índice en IA-32, la excepción es que
- EBP no puede ser registro base
 - EBP no puede ser registro índice
 - ESP no puede ser registro base
 - ESP no puede ser registro índice**
-
16. El registro SP / ESP / RSP...
- es un registro transparente al usuario y contiene la instrucción que se está ejecutando
 - es un registro de propósito específico y contiene la dirección de la cima de la pila**
 - es un registro transparente al usuario y contiene la dirección de memoria a la que se está accediendo
 - es un registro de propósito específico y contiene la dirección de la siguiente instrucción a ejecutar
-
17. Diferencias gcc Linux IA-32/x86-64: marcar la respuesta falsa
- los enteros largos (long) pasan de 32 a 64 bits
 - los punteros (void*) pasan de 32 a 64 bits
 - el tipo double pasa de 4 B a 8 B**
 - long double pasa de 10/12 B a 16 B
-
18. ¿Cuál de los siguientes fragmentos de código deja en %eax un resultado distinto a los otros tres fragmentos?



Tienda 4

Tienda 3

Tienda 2

Tienda 1

- a. `mov $-1, %edx
sub %eax, %edx
mov %edx, %eax`
- b. `not %eax
add $1, %eax`
- c. `xor %edx, %edx
sub %eax, %edx
mov %edx, %eax`
- d. `neg %eax`

19. Si A y B son dos enteros almacenados respectivamente en %eax y %ebx, ¿cuál de las siguientes implementaciones de

```
if (!A && !B) {...then part...}
```

es incorrecta?

- a. `or %ebx, %eax
jne not_true
...then part...
not_true:
...`
- b. `cmp $0, %eax
jne not_true
cmp $0, %ebx
jne not_true
...then part...
not_true:
...`
- c. `test %ebx, %eax
jne not_true
...then part...
not_true:
...`
- d. `test %eax, %eax
jne not_true
test %ebx, %ebx
jne not_true
...then part...
not_true:
...`

20. Dada la siguiente declaración en lenguaje C, una estructura de este tipo podría ocupar en un sistema Linux IA-32 o bien en uno x86-64 un total de...

```
struct a{  
    int i;  
    double d;  
    char c;  
    short s; };
```

- a. 18B
- b. 20B
- c. 22B
- d. **24B**

21. En un sistema Linux x86-64, ¿cuál de las siguientes expresiones es equivalente a la expresión C $(x[2] + 4)[3]$? Suponer que previamente se ha declarado int **x.

- a. `*((x + 16)) + 28`
- b. `*((x + 2) + 7)`
- c. `(*(*x + 2) + 4) + 3`
- d. ***((x + 2) + 4) + 3**

22. Una unidad de control microprogramada se denomina "con secuenciamiento de microinstrucciones explícito" según tenga o no tenga

- a. ROM/PLA para traducir el codop en dirección de inicio de microprograma (goto f(IR))
- b. un multiplexor para seleccionar la fuente de la dirección de la memoria de control
- c. **micro-contador de programa atacando a las líneas de dirección de la memoria de control**
- d. microcódigo de decodificación que analice el codop bit a bit de izquierda a derecha

23. Dado un camino de datos concreto, un posible formato de microprogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)

- a. codificación
- b. solapamiento
- c. **microbifurcaciones**
- d. longitud relativa de microinstrucción

24. Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señalar la respuesta falsa)

- a. registros de acopllo (coste de la segmentación)
- b. fragmentación desigual (duración desigual de etapas)
- c. riesgos (hazards)
- d. **cola de instrucciones (precaptación)**

25. La técnica de "adelanto de registros" (register forwarding) en un cauce segmentado se usa para limitar el impacto de los riesgos...

- a. estructurales
- b. organizativos
- c. de control
- d. **(por dependencias) de datos**

26. Las técnicas principales de E/S son (señalar la respuesta falsa)

- a. DMA (por acceso directo)
 - b. E/S programada
 - c. **E/S cableada (hardwired)**
 - d. IRQ (por interrupciones)
-

27. Para determinar la causa de una interrupción se pueden usar las siguientes técnicas: (señalar la respuesta falsa)

- a. múltiples líneas de interrupción INT1#, INT2#...
 - b. **línea de reconocimiento INTA#**
 - c. consulta de estado, o polling
 - d. interrupciones vectorizadas
-

28. Utilizar una cache en el mismo chip del procesador:

- a. aumenta el tamaño de los bloques enviados entre cache y procesador
 - b. **reduce los tiempos de ejecución**
 - c. reduce el tamaño del bus
 - d. aumenta la tasa de aciertos
-

29. En un sistema Linux IA-32, ¿cuántos enteros se podrían almacenar en una línea de cache, si la cache del procesador fuera de 4 KB, asociativa por conjuntos de 4-vías, y contuviera 4 conjuntos?

- a. 16
 - b. 32
 - c. **64**
 - d. 128
-

30. La cache con correspondencia directa se puede considerar como un caso límite de la asociativa por conjuntos, en donde...

- a. **solo hay 1 línea por conjunto**
 - b. solo hay 1 palabra por bloque
 - c. solo hay 1 conjunto por cache
 - d. ninguna de las anteriores
-

Nombre:
DNI:
Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿Qué hace gcc -O0?
 - a. **Compilar sin optimización**
 - b. Compilar .c→.o (fuente C a objeto)
 - c. Compilar .s→.o (fuente ASM a objeto)
 - d. Compilar .c→.s (C→ASM sin generar objeto)
 2. ¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?
 - a. **-m32**
 - b. **-m64**
 - c. **-march32**
 - d. **-march64**
 3. La etiqueta del punto de entrada a un programa ensamblador en el entorno de las prácticas 1 a 4 (GNU/as Linux x86) es:
 - a. **_main**
 - b. **.L0**
 - c. **_start**
 - d. **_init**
 4. La siguiente línea en la sección de datos de un programa en ensamblador de IA32

result: .int 0,0
 - a. Reserva espacio para un único entero, inicializado a 0,0
 - b. Reserva espacio para un entero, inicializado a 0, seguido de un dato de tamaño indefinido, también inicializado a 0
 - c. **Reserva espacio para dos enteros, inicializados ambos a 0**
- d. Reserva espacio para un único entero, inicializado a 0, en la posición de memoria 0
5. El volcado mostrado abajo se ha obtenido con el comando...

00000000 <main>:

```
0: 55          push %ebp
1: 89 e5       mov  %esp,%ebp
3: 83 e4 f0    and  $-16,%esp
6: 83 ec 10   sub  $0x10,%esp
9: c7 44 24 04 03 movl $3, 4(%esp)
e: 00 00 00
11: c7 04 24 01 00 movl $0x1, (%esp)
16: 00 00
18: e8 fc ff ff ff call <main+0x19>
1d: c9          leave
1e: c3          ret
```

 - a. objdump -h p
 - b. objdump -d p
 - c. **objdump -d p2.o**
 - d. objdump -t p2.o
 6. En la práctica "media" se desea invocar desde lenguaje ensamblador la función printf() de libC. Eso implica que este programa, como todo programa que use libC,
 - a. es obligatorio que contenga main (y entonces es más cómodo usar gcc para enlazar)
 - b. es obligatorio enlazarlo usando gcc (y entonces es más cómodo que contenga main)
 - c. es ventajoso para ensamblarlo que contenga main, y entonces es conveniente enlazarlo usando gcc (aunque ambas cosas son opcionales)

- d. es ventajoso para enlazarlo usar gcc, y entonces es conveniente que contenga main (aunque ambas cosas son opcionales)
-
7. En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. Un estudiante entrega la siguiente versión
- ```
$lista en EBX, longlista en ECX
suma:
 mov $0, %eax
 mov $0, %edx
 mov $0, %esi

bucle:
 add (%ebx,%edx,4), %eax
 jnc seguir
 inc %edx

seguir:
 inc %esi
 cmp %esi,%ecx
 jne bucle
 ret
```
- Esta función presenta una única diferencia frente a la solución recomendada en clase, relativa al indexado en el array.
- Esta función suma:
- Produce siempre el resultado correcto
  - Fallaría con lista: .int 1,1,1,1, 1,1,1,1, ...
  - Fallaría con lista: .int 1,2,3,4, 1,2,3,4, ...**
  - No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
- 
8. En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32 bits (sin signo)? Se usa notación decimal y espacios como separadores de millares/millones/etc.
- 10 000 000
  - 100 000 000
  - 1 000 000 000**
  - 10 000 000 000
- 
9. En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando desbordamiento. De entre los
- siguientes, ¿cuál es el valor negativo más pequeño (en valor absoluto) que repetido en toda la lista causaría desbordamiento con 32 bits (en complemento a 2)? Se usa notación decimal y espacios como separadores de millares/millones/etc.
- 10 000 000
  - 100 000 000**
  - 1 000 000 000
  - 10 000 000 000
- 
10. ¿Cuál es el **popcount** (peso Hamming, nº de bits activados) del número 19?
- 2
  - 3**
  - 4
  - 5
- 
11. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de **popcount3**:
- ```
int popcount3(unsigned* array,
              int len){
    int i, res = 0;
    unsigned x;
    for( i = 0; i < len; i++ ) {
        x = array[i];
        asm("ini3:          \n"
            "shr %[x]         \n"
            "adc $0, %[r]      \n"
            "add $0, %[x]      \n"
            "jne ini3          \n"
            : [r] "+r" (res)
            : [x] "r" (x) );
    }
    return res;
}
```
- Esta función sólo tiene una diferencia con la versión "oficial" recomendada en clase. En concreto, una instrucción máquina en la sección **asm()** es distinta.
- Esta función **popcount3**:
- produce siempre el resultado correcto
 - fallaría con **array={0,1,2,3}**
 - fallaría con **array={1,2,4,8}**
 - no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- Perfecto para largas noches de risas con amigos



12. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:

```
int popcount3(int* array, int len) {
    long val = 0;
    int i;
    unsigned x;
    for (i=0; i<len; i++) {
        x= array[i];
        do{
            val += x & 0x1;
            x >>= 1;
        } while (x);
        val += (val >> 16);
        val += (val >> 8);
    }
    return val & 0xFF;
}
```

Esta función es una mezcla inexplicada de las versiones "oficiales" de **popcount2** y **popcount4**, incluyendo diferencias en 2 tipos de datos, la ausencia de la variable **res** y la diferente posición de la máscara **0xFF**.

Esta función popcount3:

- produce siempre el resultado correcto
- fallaría con **array={0,1,2,3}**
- fallaría con **array={1,2,4,8}**
- no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**

13. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount4:

```
int popcount4(unsigned* array,
              int len) {
    int i,j;
    unsigned x;
    int result = 0;
    for(i=0;i<len;i++) {
        x=array[i];
        for(j=0;j<8;j++) {
            result += x & 0x01010101;
            x>>=1;
        }
    }
    result += (result >> 16);
    result += (result >> 8);
    return result & 0xFF;
}
```

}

Esta función presenta varias diferencias con la versión "oficial" recomendada en clase, incluyendo la ausencia de una variable auxiliar **val** y la diferente posición de los desplazamientos **>>** y máscara **0xFF**.

Esta función popcount4:

- produce siempre el resultado correcto
- fallaría con **array={0,1,2,3}**
- fallaría con **array={1,2,4,8}**
- no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**

14. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity5:

```
int parity5(unsigned * array,
            int len) {
    int i,j, result = 0;
    unsigned x;
    for(i = 0; i<len; i++){
        x=array[i];
        for(j=1; j<8*sizeof(int); j*=2)
            x ^= x >> j;
        result += x & 0x1;
    }
    return result;
}
```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase, en las condiciones del bucle **for** interno.

Esta función parity5:

- Produce siempre el resultado correcto**
- Fallaría con **array={0,1,2,3}**
- Fallaría con **array={1,2,4,8}**
- No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

15. La función **gettimeofday()** en la práctica de la "bomba digital" se utiliza para

- Para comparar las duraciones de las distintas soluciones del programa
- Para imprimir la fecha y hora
- Para cifrar la contraseña en función de la hora actual
- Para cronometrar lo que tarda el usuario en introducir la contraseña**

16. Un fragmento de una “bomba” desensamblada es:

```
0x0804873f: call 0x8048504 <scanf>
0x08048744: mov 0x24(%esp), %edx
0x08048748: mov 0x804a044, %eax
0x0804874d: cmp %eax, %edx
0x0804874f: je 0x8048756 <main+230>
0x08048751: call 0x8048604 <boom>
0x08048756: ...
```

La contraseña/clave en este caso es...

- a. el string almacenado a partir de la posición de memoria 0x804a044
- b. el string almacenado a partir de la posición de memoria 0x24(%esp)
- c. el entero almacenado a partir de la posición de memoria 0x804a044
- d. el entero 0x804a044

17. Una de las “bombas” utiliza el siguiente bucle para cifrar la cadena con la contraseña introducida por el usuario:

```
80485bb: rolb $0x4, (%eax)
80485be: add $0x1, %eax
80485c1: cmp %edx, %eax
80485c3: jne 80485bb <encrypt+0x20>
```

La instrucción **rolb** rota el byte destino hacia la izquierda tantos bits como indica el operando fuente. Si inicialmente eax apunta a la cadena del usuario, que se compara con otra cadena “\x16\x26\x27\x16\x36\x16\x46\x16\x26\x27\x16”, almacenada en el código, la contraseña es:

- a. “\x61\x62\x72\x61\x63\x61\x64\x61\x62\x72\x61” (“abraacadabra”)
- b. “\x61\x72\x62\x61\x64\x61\x63\x61\x72\x62\x61” (“arbadacarba”)
- c. “\x63\x61\x64\x61\x62\x72\x61\x61\x62\x72\x61” (“cadabraabra”)
- d. “\x61\x62\x72\x61\x61\x62\x72\x61\x63\x61\x64” (“abraabracad”)

18. Una de las “bombas” utiliza el siguiente código para cifrar la clave numérica introducida por el usuario y ahora almacenada en eax:

```
804870d: xor    $0xffff, %eax
8048712: mov    $0x2, %ecx
8048717: cltd
8048718: idiv   %ecx
804871a: cmp    %eax, 0x804a034
```

Si el entero almacenado a partir de 0x804a034 es 0x7ff, la clave numérica puede ser:

- a. 0x10094F97 (269045655)
- b. 0xffff (2047)
- c. 0x7ff (4095)
- d. 1

19. En el programa line.cc de la práctica 5, si para cada tamaño de línea (line) recorremos una única vez el vector, la gráfica resultante es decreciente porque:

- a. Cada vez que line aumenta al doble, el número de aciertos por localidad temporal aumenta, porque ya habíamos accedido a cada posición i del vector cuando lo recorrimos en el punto anterior del eje X.
- b. Cada vez que line aumenta al doble, el número de aciertos por localidad espacial aumenta, porque ya habíamos accedido a cada posición i-1 del vector cuando lo recorrimos en el punto anterior del eje X.
- c. Cada vez que line aumenta al doble, se accede con éxito a más posiciones del vector en niveles de la jerarquía de memoria más rápidos.
- d. Cada vez que line aumenta al doble, realizamos la mitad de accesos al vector que para el valor anterior.

20. ¿Cuál de las siguientes afirmaciones sobre el programa size.cc de la práctica 5 es cierta?

- a. La diferencia de velocidades entre L2 y L3 es mayor que la diferencia de velocidades entre L1 y L2.
- b. Si continuáramos multiplicando por 2 el tamaño del vector en el eje X obteniendo más puntos de la gráfica, esta continuaría horizontal para cualquier valor más allá de 64 MB.
- c. La gráfica tiene escalones hacia arriba porque en cada punto del eje X accedemos al mismo número de elementos del vector y el número de aciertos por localidad temporal disminuye bruscamente en ciertos puntos al aumentar el tamaño del vector.
- d. La gráfica tiene tramos horizontales porque el hecho de realizar la mitad de accesos al vector en cada punto de un tramo horizontal respecto al anterior punto de ese mismo tramo horizontal es compensado por el número de fallos creciente en ese mismo tramo horizontal.

Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Cuál de las siguientes afirmaciones sobre el benchmark SPEC CPU es falsa?
 - a. La última versión es SPEC CPU2006 V1.2 de 2011
 - b. Se cronometran unos 12 tests de enteros (CINT2006) y unos 17 tests de punto flotante (CFP2006)
 - c. Se usa como referencia un computador UltraSPARC II 300MHz, y para cada test se calcula el cociente entre el tiempo de ejecución en el computador a testear y en el de referencia
 - d. El resultado final es la media aritmética de las (12 ó 17) velocidades, bien sea de enteros ó de punto flotante (SPECint2006 ó SPECfp2006)

2. ¿En qué generación, dentro de la historia de los computadores digitales, aparecieron la microprogramación, la segmentación de cauce, la memoria cache, los S.O. multiusuario y la memoria virtual?
 - a. 2ª generación (1955-65)
 - b. 3ª generación (1965-75)
 - c. 4ª generación (1975-85)
 - d. esas innovaciones se repartieron a lo largo de varias generaciones, no sólo una

3. Respecto a tamaños de tipos integrales en x86 y x86-64, la excepción es que
 - a. int pasa de 4 B (x86) a 8 B (x86-64)
 - b. long int pasa de 4 B a 8 B
 - c. long long int pasa de 4 B a 8 B
 - d. ninguna de las anteriores

4. Con el repertorio IA32, para sumar %eax y %ebx dejando el resultado en %ecx se podría hacer lo siguiente:
 - a. lea %eax, %ebx, %ecx
 - b. lea (%eax, %ebx, 1), %ecx
 - c. lea %ecx, [%eax, %ebx]
 - d. lea %ecx, %ebx, %eax

5. Cuál de las instrucciones máquina siguientes es incorrecta en x86-64:
 - a. testl %edx, %edx
 - b. movl %r8, %eax
 - c. movl (%rdi,%rcx,4), %edx
 - d. addq \$1, %rcx

6. Si la variable val está almacenada en ebx y la variable x está almacenada en eax, la sentencia **val ^= x;** se puede traducir a ensamblador como:
 - a. xorl %ebx,%eax
 - b. xorl %eax,%ebx
 - c. andl %ebx,%eax
 - d. testl %eax,%ebx

7. Para poner a 1 el bit 5 del registro %edx sin cambiar el resto de bits podemos usar la instrucción máquina:
 - a. and \$32, %edx
 - b. and \$0x5, %edx
 - c. or \$0b101, %edx
 - d. or \$0x20, %edx

8. Si tenemos un número n , de 64 bits, almacenado en la pareja de registros EDX:EAX (EDX contiene los 32 bits más

significativos y EAX los 32 bits menos significativos) y queremos realizar la división $n / 2^{32}$ entonces:

- a. Podemos quedarnos con EDX, pero sólo en el caso de que n sea un número sin signo.
- b. **Podemos quedarnos con EDX tanto si n es un número con signo como sin signo.**

- c. Podemos usar las instrucciones siguientes, pero sólo en el caso de que n sea un número sin signo:

```
mov $0x100000000,%ecx  
div %ecx
```

- d. Podemos usar las instrucciones siguientes tanto si n es un número con signo como sin signo:

```
mov $0x100000000,%ecx  
div %ecx
```

-
- 9. ¿Dónde está ubicado el primer argumento a una función (suponer código ensamblador cdecl generado por gcc para Linux/x86) inmediatamente después de ejecutar la instrucción call?

- a. %ebp + 0x4
- b. %ebp - 0x4
- c. **%esp + 0x4**
- d. %esp - 0x4

-
- 10. Dado el código C siguiente:

```
struct data {  
    char str[16];  
};  
char *f(struct data *ptr) {  
    return &(ptr->str[2]);  
}
```

la función se traducirá a ensamblador de x86-64 como:

- a. **leaq 2(%rdi), %rax**
ret
- b. movq (%rdi,2), %rax
ret
- c. movq 2(%rdi), %rax
ret
- d. leaq (%rdi,2), %rax
ret

-
- a. en x86 Linux alinea double a 4x (Windows no)

- b. en x86 Linux alinea long double a 4x (Windows también)

- c. **en x86-64 Linux alinea double a 8x** (Windows también)

- d. **en x86-64 Linux alinea float a 8x** (Windows **11-D**) también

-
- 12. Si la estructura **struct a** ocupa un espacio de 28 bytes en memoria, ¿cuántos bytes ocupa la siguiente estructura **struct b** cuando se compila en 64 bits?

```
struct b {  
    struct a a1;  
    int i;  
    struct a a2;  
};
```

- a. 24 bytes
- b. **60 bytes**
- c. **64 bytes**
- d. 84 bytes

-
- 13. Respecto a los términos microinstrucción y microcódigo:

- a. Son equivalentes, llamamos microcódigo o microinstrucción a una palabra de la memoria de control
- b. Una microinstrucción está programada en microcódigo, que es un lenguaje para programar señales de control
- c. Un microcódigo controla una serie de señales de control relacionadas (por ejemplo, el código 000 para que la ALU realice la suma), y varios microcódigos juntos forman una microinstrucción
- d. **Microcódigo es el contenido de la memoria de control, y una microinstrucción es una palabra de dicha memoria**

-
- 14. ¿Cuál de las siguientes afirmaciones es verdadera?

- a. **La unidad de control necesita como entrada el registro de estado para poder controlar la ejecución de las instrucciones de salto condicional.**
- b. El registro de instrucción es un registro de propósito específico que contiene la dirección de la siguiente instrucción a ejecutar.

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- **Perfecto para largas noches de risas con amigos**



- c. Las únicas instrucciones en las que algunas de sus fases de ejecución llevan un acceso a memoria son las instrucciones load y store.
- d. El registro puntero de pila es un registro de propósito general que suele contener tanto direcciones como datos.
-
- 15.** Un procesador con una unidad de control microprogramada tiene una memoria de control de 300 palabras de 100 bits, de las que 200 son diferentes. Si se rediseñara como unidad de control nanoprogramada, ¿qué tamaño ocuparía la nanomemoria que contiene las microinstrucciones completas sin repeticiones?
- 20000 bits
 - 21600 bits
 - 22400 bits
 - 30000 bits
-
- 16.** En el pseudocódigo usado para representar las microinstrucciones, la expresión “goto f(IR)”:
- Se utiliza para realizar un microsalto condicional en función del registro de estado.
 - Realiza una llamada a una microsubrutina.
 - Salta a una dirección de memoria de control que depende de la instrucción máquina actual.**
 - Permite saltar a la dirección de memoria de control del principio de un microbucle.
-
- 17.** Respecto a la predicción de saltos, alguna de las siguientes afirmaciones es falsa
- si se toma la misma decisión para cada tipo de instrucción, se trata de "predicción estática"
 - si la predicción cambia según la historia de ejecución del programa, se trata de "predicción dinámica"
 - para predicción estática, es conveniente decidir que los saltos hacia adelante siempre se cumplen, y hacia atrás no**
 - para predicción dinámica, existen, entre otros, algoritmos de dos o cuatro estados, que requieren 1 o 2 bits por instrucción
-
- 18.** Respecto a los conceptos de procesamiento segmentado y superescalar, una de las siguientes afirmaciones es falsa
- idealmente, con el segmentado se intenta ejecutar una instrucción por ciclo, y con el superescalar más de una por ciclo (al combinarlo con segmentado)
 - en cualquier procesador resulta ventajoso usar una cola de instrucciones, pero es más importante para uno segmentado (fundamental) que para uno superescalar (conveniente)**
 - por definición, un procesador superescalar debe tener varias unidades funcionales (más de una)
 - implícitamente, se presupone que un procesador superescalar emitirá más de una instrucción por ciclo
-
- 19.** Respecto a los conceptos de interfaz de dispositivo, controlador(a), puerto de E/S:
- La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo**
 - Cada puerto o interfaz es una línea de comunicación con el procesador. El conjunto de ellos forma el controlador.
 - El puerto, o interfaz, contiene los controladores necesarios para comunicar el dispositivo con el procesador
 - El interfaz contiene las controladoras necesarias para conectar los puertos con el procesador
-
- 20.** Respecto a los conceptos de procesador de E/S, canal de E/S, dispositivos de E/S:
- Un procesador o canal tiene un repertorio de instrucciones específico para manejar los dispositivos E/S**
 - Cada canal es una línea de comunicación entre el procesador y un dispositivo de E/S.
 - Al conjunto de conexiones entre el procesador y los dispositivos se le denomina canal de E/S (de ese ordenador)
 - La pregunta es capiosa, el procesador no es E/S, son otros dos componentes von Neumann distintos (ALU+UC)
-
- 21.** La E/S programada:
- Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU tiene el control de toda la operación.
 - Mejora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU es más rápida que el controlador de interrupciones y la interfaz del periférico.
 - Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque una

instrucción de transferencia individual de datos con la interfaz del periférico (por ej. IN, OUT) es más lenta en E/S programada que en E/S por interrupciones.

- d. Empeora las prestaciones globales del sistema respecto a la E/S por interrupciones porque la CPU debe encargarse de la sincronización con la interfaz del periférico haciendo una espera activa.

22. Una puerta AND con 16 entradas conectada a un bus de direcciones de 16 bits, con todos los bits negados excepto A10 y A6, permite seleccionar un dispositivo (con CS activa en alta) en la dirección:

- a. 0xFDDF
b. 0xFBFF
c. 0x0220
d. 0x0440

23. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de 2^{13} palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?

- a. 2^{10}
b. 2^{12}
c. 2^{11}
d. 2^{13}

24. Un procesador accede en el instante de tiempo t a una posición de memoria d(t). Poco tiempo después (en el instante de tiempo t+k) accede a la posición anterior d(t)-1. Esos dos accesos son un ejemplo de...

- a. Localidad espacial
b. Localidad temporal
c. No tiene nombre, ese tipo de localidad con incremento negativo (d(t)-1) no se ha estudiado en clase
d. No es una localidad, esa condición no guarda relación con el concepto de localidad

25. Una jerarquía de memoria consta de una cache de con una tasa de aciertos del 92% y 4 ns de tiempo de acceso y una memoria principal con una tasa de aciertos del 100% y 100 ns de tiempo de acceso. ¿Cuál es el tiempo promedio estimado de acceso a memoria?

- a. 6 ns

- b. 8 ns
c. 10 ns
d. 12 ns

26. Una SRAM de 1Mx4bit (4Mbit) puede venir organizada en 2048 filas, dedicando por tanto al decodificador de columnas...

- a. 6 bits
b. 7 bits
c. 8 bits
d. 9 bits

27. Un sistema basado en un microprocesador con un bus de datos de n bits y un bus de direcciones de 16 bits direcciona la memoria por palabras de n bits y dispone de una memoria SRAM formada por dos módulos de 16 K x n cada uno. ¿Qué porcentaje del mapa de memoria está ocupado por la SRAM?

- a. 12,5%
b. 25%
c. 50%
d. 100%

28. Un módulo de memoria de 16 GB está formado por varios chips DRAM de 1024Mx4. ¿Cuántos chips DRAM necesita el módulo?

- a. 4
b. 8
c. 16
d. 32

29. Una cache de 256 B asociativa por conjuntos de 4-vías con líneas de 16 B tendría

- a. 4 conjuntos
b. 16 conjuntos
c. 64 conjuntos
d. ningún conjunto

30. En un sistema con memoria de bytes, ¿cuál sería el tamaño de una línea de cache, si la cache del procesador fuera de 4MB, asociativa por conjuntos de 16-vías, y contuviera 4096 conjuntos?

- a. 16 B
b. 32 B
c. 64 B
d. 128 B



Nombre:

DNI:

Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿Cuál de las siguientes instrucciones máquina copia en EAX la dirección efectiva resultante de la operación EDX*4 + EBX?
- a. leal 4(%edx, %edx), %eax
 - b. leal (%ebx, %edx, 4), %eax**
 - c. movl 4(%edx, %edx), %eax
 - d. movl (%ebx, %edx, 4), %eax

2. ¿Cuál de las siguientes instrucciones máquina copia en EAX el entero almacenado en la posición de memoria cuya dirección efectiva es el resultado de la operación EDX*4 + EBX?
- a. leal 4(%edx, %edx), %eax
 - b. leal (%ebx, %edx, 4), %eax
 - c. movl 4(%edx, %edx), %eax
 - d. movl (%ebx, %edx, 4), %eax**
3. Los switches `-m elf_i386` y `-m elf_x86_64` para trabajar en 32 bits / 64 bits corresponden a la herramienta...
- a. gcc
 - b. as
 - c. ld**
 - d. nm

4. Si ECX vale 0, la instrucción adc \$0,%ecx
- a. Pone CF=1
 - b. Pone CF=0**
 - c. Cambia CF
 - d. No cambia CF

5. Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,  
       0x10000000, 0x20000000  
longlista: .int (. lista ) / 4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\0"
```

La instrucción para copiar la dirección de memoria donde comienza lista en el registro EBX es:

- a. movl lista, %ebx
- b. movl \$lista, %ebx**
- c. movl (lista), %ebx
- d. movl \$lista, (%ebx)

6. Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,  
       0x10000000, 0x20000000  
longlista: .int (. lista ) / 4  
resultado: .quad 0x123456789ABCDEF  
formato: .ascii "suma=%llu=%llx hex\n\0"
```

la instrucción `movl longlista, %ecx` copia el siguiente valor:



- a. 4
 - b. 8
 - c. 16
 - d. 32
-

7. Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,
        0x10000000, 0x20000000
longlista: .int (. - lista) / 4
resultado: .quad 0x123456789ABCDEF
formato: .ascii "suma=%llu=%llx hex\n\0"
```

y suponiendo que hemos llamado a una función *suma* que devuelve un número de 64 bits en la pareja EDX:EAX, las instrucciones que copian ese número en *resultado* son:

- a. movl %eax, resultado
movl %edx, resultado+4
 - b. movl (%eax), resultado
movl (%edx), resultado+4
 - c. movl %eax, resultado+4
movl %edx, resultado
 - d. movl (%eax), resultado+4
movl (%edx), resultado
-

8. Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,
        0x10000000, 0x20000000
longlista: .int (. - lista) / 4
resultado: .quad 0x123456789ABCDEF
formato: .ascii "suma=%llu=%llx hex\n\0"
```

la llamada correcta a *printf* será:

- a. push resultado+4
push resultado
push \$formato
call printf
add \$12, %esp
 - b. push resultado+4
push resultado
push resultado+4
push resultado
-

```
push $formato
call printf
add $20, %esp
```

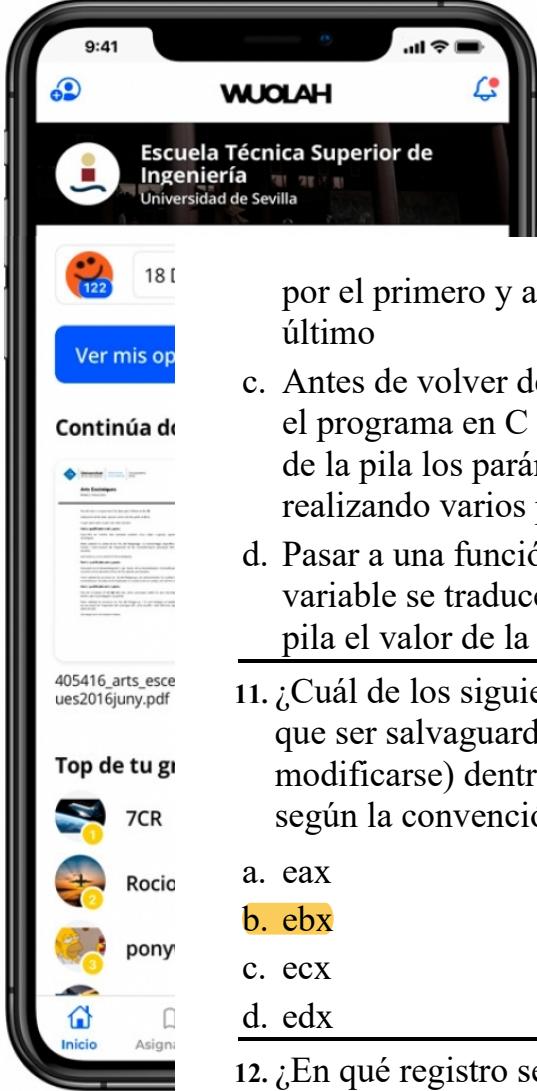
- c. push resultado
push resultado+4
push \$formato
call printf
add \$12, %esp
 - d. push resultado
push resultado+4
push resultado
push resultado+4
push \$formato
call printf
add \$20, %esp
-

9. En la práctica “media” se pide sumar una lista de enteros ***con*** signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en los ***dos*** primeros elementos de la lista causaría overflow con 32 bits al realizar la suma de ***esos dos*** primeros elementos de la lista?

- a. 0x0400 0000
 - b. 0x0800 0000
 - c. 0x4000 0000
 - d. 0x8000 0000
-

10. ¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?

- a. En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
- b. Los parámetros se introducen en la pila en el orden en el que aparecen en la llamada de C, es decir, empezando



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- por el primero y acabando por el último
- Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop
 - Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la variable

-
11. ¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de una subrutina según la convención cdecl para IA32?
- eax
 - ebx**
 - ecx
 - edx
-
12. ¿En qué registro se pasa el primer argumento a una función en Linux gcc x86-64?
- ecx
 - edx
 - esi
 - edi**
-

13. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount3:

```
int popcount3(unsigned* array,
              int len) {
    int i, res = 0;
    unsigned x;
    for( i=0; i<len; i++ ) {
        x = array[i];
        asm("ini3:          \n"
             "shr %[x]       \n"
             "adc $0, %[r]   \n"
             "add $0, %[x]   \n"
             "jne ini3      \n"
             : [r] "+r" (res)
             : [x] "r" (x) );
    }
    return res;
}
```

Esta función produce siempre el resultado correcto, a pesar de que una instrucción máquina en la sección **asm()** es distinta a la que se esperaba después de haber estudiado **pcount_r** en teoría. La instrucción distinta también se podría haber cambiado por...

- sar %[x]
 - adc \$0, %[x]
 - test %[x], %[x]**
 - cmp %[x], %[r]
-
14. En la práctica de la bomba necesitamos estudiar el código máquina de la bomba del compañero. A veces dicho código no se visualiza directamente en el depurador ddd, y algunas de las técnicas que se pueden probar para conseguir visualizarlo son... (marcar la opción ***falsa***)

- comprobar que está activado el panel *View → Machine Code Window*
 - escribir **info line main** en el panel de línea de comandos gdb
 - recompilar con información de depuración, por si se nos había olvidado, ya que sin -g el ejecutable no contiene información de depuración**
 - asegurarse de que se ha escrito correctamente el nombre del ejecutable
-

15. En la práctica de la bomba, el primer ejercicio consistía en “saltarse” las “explosiones”, para lo cual se puede utilizar...

a. objdump o gdb

b. **gdb o ddd**

c. ddd o hexedit

d. hexedit u objdump

16. En la práctica de la bomba, el segundo ejercicio consistía en crear un ejecutable sin “explosiones”, para lo cual se puede utilizar...

a. objdump o gdb

b. **gdb o ddd**

c. **ddd o hexedit**

d. hexedit u objdump

17. En la práctica de la bomba, el tercer ejercicio consistía en usar un editor hexadecimal para crear un ejecutable sin “explosiones”. Para saber qué contenidos del fichero hay que modificar, se puede utilizar... (marcar la opción falsa)

a. objdump

b. gdb

c. ddd

d. hexedit

18. Suponer una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponer que para la siguiente secuencia de direcciones enviadas a la cache: 0, 2, 4, 8, 16, 32, la tasa de acierto es 0,33. ¿Cuál es el tamaño de bloque de la cache?

a. 4 bytes

b. 8 bytes

c. 16 bytes

d. Ninguno de los anteriores

19. Abajo se ofrece el listado de una función para multiplicar matrices C = A x B.

```
void mult_matr(float A[N][N],  
float B[N][N], float C[N][N]) {  
/*Asume val.ini. C={0,0...}*/  
int i,j,k;  
for (i=0; i<N; i++)  
    for (j=0; j<N; j++)  
        for (k=0; k<N; k++)  
            C[i][j] += A[i][k]*B[k][j];  
}
```

Suponer que:

- El computador tiene una cache de datos de 8 MB, 16-vías, líneas de 64 bytes.
- N es grande, una fila o columna no cabe completa en cache.
- El tamaño de los tipos de datos es como en IA32.
- El compilador optimiza el acceso a C[i][j] en un registro.

Aproximadamente, ¿qué tasa de fallos se podría esperar de esta función para valores grandes de N?

a. 1/2

b. 1/4

c. 1/8

d. 1/16

20. Sea un computador de 32 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:

```
int v[262144];  
for (i=0; i<262144; i+=8)  
v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?

a. 0 (ningún fallo)

b. 1/2 (mitad aciertos, mitad fallos)

c. 1/8 (un fallo por cada 8 accesos)

d. 1 (todo son fallos)



Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.
Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. Respecto a direccionamiento a memoria en ensamblador IA32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA.
¿Cuál?

- a. El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
- b. EBP no se puede usar como registro base
- c. ESP no se puede usar como registro índice
- d. El factor de escala S puede ser 1, 2, 4, 8

2. La extensión de signo a m bits de un número original N de n bits, con $m > n$, consiste en:

- a. Realizar la operación $2^m - N$
- b. Realizar la operación $2^m - N - 1$
- c. Incrementar la cantidad de bits a m preservando el signo y el valor del número.
- d. Incrementar la cantidad de bits a m rellenando con unos por la izquierda.

3. En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?

- a. sub %edi, %edi
adc \$0xFFFFFFFF, %edi
- b. mov \$-1, %edi
- c. mov \$-1, %edi
add \$0, %edi

- d. mov \$0, %edi
sub \$1, %edi

4. Si %rsp vale 0xdeadbeefdeadd0d0, ¿cuál será su nuevo valor después de que se ejecute pushq %rbx?

- a. 0xdeadbeefdeadd0d4
- b. 0xdeadbeefdeadd0d8
- c. 0xdeadbeefdeadd0cc
- d. 0xdeadbeefdeadd0c8

5. ¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una función long int al terminar ésta?

- a. La instrucción RET lo almacena en un registro especial de retorno.
- b. Por convención se guarda en %eax.
- c. Se almacena en pila justo encima de los argumentos de la función.
- d. Ninguna de esas formas es la correcta.

6. Comparando las convenciones de llamada de gcc Linux IA32 con x86-64 respecto a registros

- a. En IA32 %ebx es salva-invocante, pero en x86-64 %rbx es salva-invocado
- b. En IA32 %ecx es salva-invocante, y en x86-64 %rcx es salva-invocante también
- c. En IA32 %esi es salva-invocado, y en x86-64 %rsi es salva-invocado también
- d. En IA32 %ebp es especial (marco de pila), y en x86-64 %rbp también

7. Son funciones de la unidad de control:



- a. la codificación de las instrucciones máquina
 - b. la lectura de memoria principal de la instrucción apuntada por el μ PC
 - c. el secuenciamiento de las instrucciones máquina
 - d. todas las respuestas son ciertas
-

8. Respecto a MBR y MAR

- a. Ambos son accesibles por el programador
 - b. MAR contiene el dato/instrucción que se leerá o escribirá en memoria
 - c. MAR requiere menos señales de control que MBR
 - d. Ambos permiten guardar información sobre el marco de pila
-

9. Una instrucción máquina puede desglosarse en las siguientes operaciones elementales:

$sp := sp - 1; m[sp] := pc; pc := x$

Probablemente se trate de una instrucción de:

- a. apilamiento
 - b. llamada a subrutina
 - c. carga local
 - d. almacenamiento local
-

10. En una unidad de control micropogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de

- a. 4 bits
 - b. 5 bits
 - c. 16 bits
 - d. 17 bits
-

11. Dado un camino de datos concreto, un posible formato de micropogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta falsa)

- a. codificación
 - b. solapamiento
-

- c. microbifurcaciones
 - d. longitud relativa de microinstrucción
-

12. El control residual se utiliza para:

- a. reducir el tiempo de ejecución de las instrucciones máquina
 - b. eliminar los bits residuales de la ejecución de las microinstrucciones
 - c. reducir el tamaño de la memoria de control
 - d. ninguna de las anteriores es cierta
-

13. Un procesador está segmentado en las etapas F, D, E, M y W. Cada una de ellas consume un tiempo t . La aceleración ideal (si no hay riesgos) al ejecutar n instrucciones respecto a un procesador no segmentado será:

- a. $5n / (4+n)$
 - b. $(4+n) / 5t$
 - c. $4n / (5+n)$
 - d. $(5+n) / 4t$
-

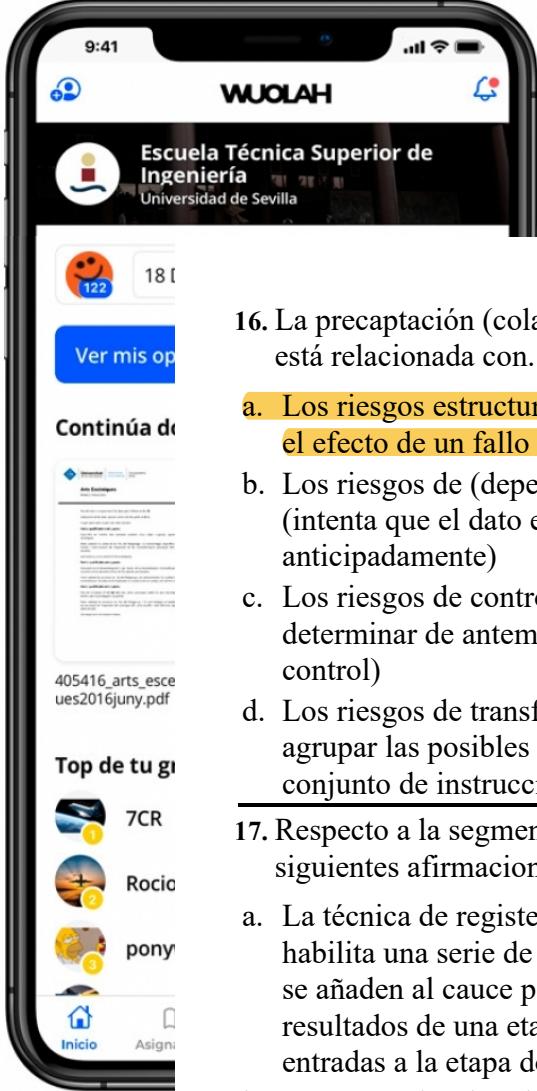
14. En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:

- a. Un incremento de las prestaciones
 - b. Un mayor retraso en la ejecución de los programas debido al incremento del número de etapas
 - c. Una disminución en la posible dependencia de datos
 - d. Una disminución de la máxima frecuencia de reloj a la que puede operar el cauce
-

15. En la secuencia de instrucciones siguiente, siendo el primer registro el destino, ¿cuántos riesgos se dan?

sub r2,r1,r3
or r8,r6,r2

- a. Un riesgo estructural
 - b. Un riesgo por dependencia de datos
 - c. Un riesgo estructural y dos por dependencia de datos
 - d. Dos riesgos por dependencia de datos y uno de control
-



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

16. La precaptación (cola de instrucciones) está relacionada con...

- a. Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
- b. Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
- c. Los riesgos de control (intenta determinar de antemano el flujo de control)
- d. Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)

17. Respecto a la segmentación, ¿cuál de las siguientes afirmaciones es falsa?

- a. La técnica de register forwarding habilita una serie de caminos (buses) que se añaden al cauce para permitir que los resultados de una etapa pasen como entradas a la etapa donde son necesarias
- b. La reorganización del código y la introducción de instrucciones nop permite evitar dependencias de datos
- c. Retrasar la fase de decisión saltar/no saltar de las instrucciones de salto condicional contribuye a mejorar el rendimiento del procesador
- d. Cuantas más etapas tenga un cauce, más instrucciones se estarán ejecutando en distintas fases y más posibilidades se presentan de que existan riesgos entre ellas

18. ¿Cuál de los siguientes modos de direccionamiento es menos preferible para un procesador de 32 bits y con tamaño de instrucción de 32 bits?

- a. registro
- b. indexado
- c. indirecto a través de registro
- d. directo (o absoluto)

19. La conexión entre un dispositivo de E/S y el procesador mediante bus:

- a. Es difícil de expandir
- b. Permite conectar en paralelo varios dispositivos

- c. Requiere mucha circuitería
- d. Requiere multiplexores y demultiplexores para las señales de datos

20. El fragmento de código ensamblador de un microprocesador de 8 bits

```
lds IOBuf    ; Apuntar puntero pila a
              ; ...área mem.intermedia
idx Count   ; Inicializar X-contador
poll lda a Status; Leer estado en A
bpl poll    ; Signo(A)!=1 => repetir
lda a Data  ; Leer dato en A
psh a       ; Transferir dato a pila
dex         ; Decrementar contador X
bne poll    ; Seguir leyendo si X!=0
```

corresponde a:

- a. Entrada programada con consulta de estado
- b. Salida programada sin consulta de estado
- c. Entrada programada sin consulta de estado
- d. Salida programada con consulta de estado

21. En la E/S controlada por interrupciones:

- a. El controlador de DMA transfiere bloques de datos por el bus del sistema.
- b. El controlador de DMA envía una petición de interrupción a la CPU.
- c. La CPU lee y comprueba el estado de los dispositivos de E/S (en el caso de consulta de estado).
- d. La CPU transfiere el control a una rutina de servicio cuando recibe una interrupción.

22. La instrucción máquina DI (Disable Interrupts), conocida como CLI (Clear Interrupt Flag) en x86, se utiliza para desactivar:

- a. Todas las interrupciones enmascarables
- b. Las interrupciones de inferior o igual prioridad a una dada
- c. Determinados niveles de interrupción de forma selectiva
- d. Las interrupciones software

- 23.** Con nueve controladores de interrupciones 8259 se pueden manejar exactamente:
- 8 niveles de prioridad
 - 16 niveles de prioridad
 - 24 niveles de prioridad
 - Ninguna de las anteriores es cierta
-
- 24.** ¿Cuál de los siguientes es un registro de un controlador de DMA?
- IR (Instruction Register)
 - PC (Program Counter)
 - SP (Stack Pointer)
 - WC (Word Count)
-
- 25.** Respecto al refresco de memorias DRAM, ¿cuál de las siguientes afirmaciones es falsa?
- Una operación de refresco consiste en dar un impulso /CAS junto con una dirección de columna.
 - Los chips DRAM refrescan automáticamente la fila accedida en cualquier ciclo de lectura o escritura.
 - Se precisa una circuitería auxiliar, externa al chip DRAM o integrada en él, que produzca ciclos de refresco.
 - Los ciclos de refresco deben producirse cada pocos ms (milisegundos).
-
- 26.** La tasa de aciertos A_i del nivel i de una jerarquía de memoria no depende de:
- La capacidad (tamaño) s_i del nivel i .
 - La estrategia de administración de memoria.
 - La unidad de la transferencia de información x_i entre el nivel i y el $i+1$.
 - El ancho de banda b_i del nivel i .
-
- 27.** La política de correspondencia de una memoria cache con 1 único conjunto es:
- Directa
 - Totalmente asociativa
 - Asociativa por conjuntos con una única línea
 - Asociativa por conjuntos de una única vía
-
- 28.** La política de correspondencia de una memoria cache con la mitad de conjuntos que líneas es:
- Asociativa por conjuntos de 2 vías
 - Totalmente asociativa de media vía
 - Asociativa por conjuntos con 2 líneas
 - Directa con 2 líneas
-
- 29.** Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta
- 8 pastillas
 - 16 pastillas
 - 32 pastillas
 - 64 pastillas
-
- 30.** Para diseñar una memoria con ancho de palabra $k*m$ (y mismo nº palabras que los módulos) a partir de módulos con ancho de palabra m , se utilizan k módulos
- repartiendo las líneas de datos entre los k módulos: el primero se conecta a $D_0 \dots D_{k-1}$, el segundo a $D_k \dots D_{2k-1}$, etc
 - repartiendo las líneas de dirección: el 1º se conecta a $A_0 \dots A_{k-1}$, el 2º a $A_k \dots A_{2k-1}$, etc
 - repartiendo líneas datos: el 1º se conecta a $D_0 \dots D_{m-1}$, el 2º a $D_m \dots D_{2m-1}$, etc
 - repartiendo líneas dirección: el 1º se conecta a $A_0 \dots A_{m-1}$, el 2º a $A_m \dots A_{2m-1}$, etc

Nombre:
DNI:
Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿Cuál de los siguientes fragmentos es correcto para comenzar un programa en ensamblador que conste de un solo archivo .s?
 - a. `.text`
`_start:`
 - b. `.text`
`.local _start`
`_start:`
 - c. `.text`
`.start _global`
`_start:`
 - d. `.text`
`.global _start`
`_start:`

2. Suponga una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponga que para la siguiente secuencia de direcciones enviadas a la cache: 1, 2, 4, 8, 16, 32, la tasa de acierto es 0,333. ¿Cuál es el tamaño de línea de la cache?
 - a. 4 bytes
 - b. **8 bytes**
 - c. 16 bytes
 - d. 32 bytes

3. En la convención cdecl estándar para arquitecturas x86 de 32 bits, cuál de las siguientes afirmaciones es cierta:
 - a. Los 6 primeros parámetros se pasan a través de registros
 - b. Solamente es necesario salvar el registro EAX

4. ¿Cuál es el popcorn (peso Hamming, nº de bits activados) del número 0x10101010?
 - a. **4**
 - b. 8
 - c. 16
 - d. 32

5. Compilar de fuente C a ejecutable usando sólo as y ld, sin gcc...
 - a. Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
 - b. Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C
 - c. Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C
 - d. **No se puede**

6. La función gettimeofday() en la práctica de popcorn y parity se utiliza para
 - a. **Comparar las duraciones de las distintas soluciones del programa**
 - b. Imprimir la fecha y hora
 - c. Cifrar el código en función de la hora actual
 - d. Cronometrar lo que tarda el usuario en pulsar una tecla

7. ¿Cuál de los siguientes registros tiene que ser salvaguardado (si va a modificarse) dentro de

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- Perfecto para largas noches de risas con amigos



una subrutina según la convención cdecl para IA32?

- ECX
- EAX
- EBP**
- EDX

8. ¿Qué hace gcc -O1?

- Compilar .s→.o (fuente ASM a objeto)
- Compilar con optimización**
- Compilar .c→.o (fuente C a objeto)
- Compilar .c→.s (C→ASM sin generar objeto)

9. Dada la siguiente definición de datos:

```
lista: .int 0x10000000, 0x50000000,
        0x10000000, 0x20000000
longlista: .int (. - lista) / 4
resultado: .quad 0x123456789ABCDEF
formato: .ascii "%llu=%llx hex\n\n\0"
```

La instrucción para copiar la dirección de memoria donde comienza lista en el registro EBX es:

- movl lista, %ebx
- movl \$lista, %ebx**
- movl (lista), %ebx
- movl \$lista, (%ebx)

10. En la práctica "media" se programa la suma de una lista de 32 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x8000 0000, ¿en qué se diferencian los resultados de ambos programas?

- no se diferencian
- en uno ocupa 32 bits, en otro 64 bits
- en uno los 16 bits superiores son 0xFFFF, en el otro no**
- en uno los 16 bits inferiores son 0x0000, en el otro no

11. En la práctica "media" se pide sumar una lista de 32 enteros CON signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando desbordamiento. Un estudiante entrega un programa que se diferencia de la versión recomendada en el siguiente bucle, en particular en la instrucción adc.

```
bucle:
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %eax, %ebp
```

```
inc %esi
cmp %esi, %ecx
jne bucle
```

Esta versión de la suma CON signo

- produce siempre el resultado correcto
- fallaría con lista: .int 1, 1, 1, 1, ...**
- fallaría con lista: .int -1,-1,-1,-1, ...
- no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

12. En la práctica "media" un estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

```
bucle:
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %edx, %ebp
    jnc nocarry
    inc %edx
nocarry:
    inc %esi
    cmp %esi, %ecx
    jne bucle
```

Este código es una mezcla de las soluciones recomendadas para suma sin signo y para suma con signo. Estando bien programado todo lo demás, este código

- produce siempre el resultado correcto**
- fallaría con lista: .int 0,1,2,3
- fallaría con lista: .int -1,-2,-4,-8
- no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

13. ¿Cuál es el popcount (peso Hamming, nº de bits activados) de una lista de N números inicializada con los valores 0..N-1?

- (N-1)*N/2
- N*(N+1)/2
- si N es par, N*(N/2)
- si N es potencia de 2, log_2(N)*N/2**

14. ¿Cuál es la paridad (XOR "lateral" de todos los bits) del número 199?

- 0
- 1**
- 2
- 4

15. Comparando los popcounts (pop(199) vs. pop(99)) y paridades (par(199) vs. par(99)) de los números 199 y 99, se verifica que
- pop(199) > pop(99)**
 - par(199) < par(99)
 - pop(199) < par(99)
 - par(199) > pop(99)
16. ¿Cuál es la suma de paridades (suma de los XOR "laterales" de los bits de cada número) de una lista de N números inicializada con los valores 0..N-1?
- (N-1)*N/2
 - N*(N+1)/2
 - si N es par, N/2**
 - si N es potencia de 2, $(\log_2(N)*N)/2$
17. En la práctica “popcount/paridad”, para cronometrar sistemáticamente las diversas versiones necesitamos una función crono() a la que se le pueda pasar como argumento cuál versión queremos cronometrar. En lenguaje C esto se puede hacer con punteros a funciones. Sabiendo que todas las versiones devuelven un valor entero, el prototipo de la función crono() debería ser:
- void crono(int * func (), char* msg);
 - void crono(int (* func)(), char* msg);**
 - void crono((int *)func (), char* msg);
 - void crono(int * func , char* msg);
18. Para corregir la práctica “bomba digital”, un profesor dispone de 26 ejecutables (y la lista de claves correspondientes) para asignar en el Laboratorio una bomba distinta a cada estudiante. Cuando un estudiante diga que la ha resuelto el profesor exigirá ver al estudiante ejecutando la bomba y tecleando la contraseña y el pin correctos para comprobar que no explota, para así anotarla como resuelta y que le puntúe al estudiante.
- Un estudiante (usando ordenador del Laboratorio con Ubuntu 10.04) dice que ha resuelto su bomba, y cuando el profesor pide que se tecleen las claves, el ddd se “bloquea” y le empieza a “parpadear” al estudiante. Para poder puntuar, lo más recomendable para el estudiante sería...
- teclear las claves (contraseña y pin), aunque ddd esté “bloqueado” y “parpadeando”
 - probar en orden los remedios básicos: pulsar <Ctrl>-C varias veces, pulsar <Enter> repetidamente, comprobar que está seleccionada “Machine Code Window”, teclear “info line main”, y si todo falla, ejecutar “`rm -rf ~/ .ddd`”
 - matar la ventana ddd, abrir un terminal con un shell, ejecutar la bomba desde línea de comandos y teclear las claves**
 - reinstalar un paquete ddd más actualizado usando “`sudo apt-get install`”
19. Suponer el mismo contexto de la pregunta anterior, donde el profesor tiene una lista de las 26+26 claves (contraseñas y pines)
- Un estudiante dice que ha resuelto su bomba, y cuando el profesor pide que se tecleen las claves, la contraseña coincide con la que tiene anotada el profesor, pero el pin no, y de todas formas la bomba no explota. Debería hacerse lo siguiente:
- la bomba tiene que puntuarle al estudiante porque no ha explotado
 - el profesor puede pedirle que vuelva a descargar la bomba original e intente repetir con ese ejecutable las claves que acaba de teclear**
 - la bomba debe marcarse como inválida y no hay que hacer más comprobaciones
 - no puede suceder lo que dice el enunciado, y en ningún caso el profesor tiene derecho a hacer comprobaciones adicionales como pedir que se vuelva a descargar la bomba
20. En la práctica de la cache, el código de “line.cc” incluye la sentencia
- ```
for (unsigned long long line=1;
 line<=LINE; line<<=1) { ... }
```
- ¿Qué objetivo tiene la expresión `line<<=1`?
- salir del bucle si el tamaño de línea se volviera menor o igual que 1 para algún elemento del vector
  - duplicar el tamaño del salto en los accesos al vector respecto a la iteración anterior**
  - volver al principio del vector cuando el índice excede la longitud del vector
  - sacar un uno (1) por el stream line

|         |
|---------|
| Nombre: |
|---------|

|      |
|------|
| DNI: |
|------|

|        |
|--------|
| Grupo: |
|--------|

### Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

1. ¿Cuál es el valor mínimo (más negativo) que puede tomar un entero de 32 bits en complemento a dos?
    - a.  $-2^{32}$
    - b.  $-2^{32} + 1$
    - c.  $-2^{31}$
    - d.  $-2^{31} + 1$

---

  2. Un datapath con bus de direcciones de 32 bits y bus de datos de 16 bits tiene un registro de 16 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro
    - a. IR
    - b. SP
    - c. MAR
    - d. PC

---

  3. ¿Cuál de las siguientes características sobre RISC es \*FALSA\*?
    - a. Para acelerar un procesador RISC se deberían emplear técnicas de segmentación.
    - b. Las instrucciones máquina en un procesador RISC deberían ser complejas y potentes.
    - c. La decodificación de las instrucciones debe ser simple: un procesador RISC debería emplear pocos formatos de instrucción.
    - d. La unidad de control de un procesador RISC debería ser cableada, no microprogramada.

---

  4. ¿Cuál de las siguientes instrucciones máquina copia en el registro EDX la dirección efectiva resultante de la operación  $EAX*8 + EBX$ ?
- a. movl (%ebx, %eax, 8), %edx
  - b. movl 8(%edx, %eax), %edx
  - c. leal (%ebx, %eax, 8), %edx
  - d. leal 8(%edx, %eax), %edx
- 
5. En el contexto general del lenguaje máquina, el acrónimo ISA suele referirse a:
    - a. Internal Standard Architecture
    - b. Integrated Set Assembly
    - c. Instruction System Architecture
    - d. Instruction Set Architecture

---

  6. En una suma de dos números en complemento a dos, se produce desbordamiento cuando
    - a. Sumamos dos positivos y el resultado es negativo o bien sumamos dos negativos y el resultado es positivo.
    - b. Sumamos dos positivos y el resultado es positivo.
    - c. Sumamos un número positivo y uno negativo.
    - d. Sumamos dos negativos y el resultado es negativo.

---

  7. Usando el repertorio IA32, para intercambiar el valor de 2 registros se pueden usar...
    - a. 4 mov, no menos (debido a la arquitectura R/M)
    - b. 3 mov, no menos (se le llama "intercambio circular")
    - c. dos instrucciones mov
    - d. una instrucción mov y una instrucción lea

8. Al ejecutar el fragmento de código:

```
leal -1(%eax), %edx
cmpb $9, %edx
ja .L2
```

se salta a .L2 si el contenido del registro %eax:

- a. es menor o igual que 1
- b. es mayor o igual que 10
- c. está fuera del intervalo [1,10]
- d. está dentro del intervalo [1,10]

9. ¿Cuál de las siguientes instrucciones convierte %eax = 5 \* %eax?

- 1) `mov 4(%eax, %eax), %eax`
- 2) `lea 4(%eax, %eax), %eax`

- a. Sólo la 2
- b. Sólo la 1
- c. Ambas, la 1 y la 2
- d. Ninguna de las dos

10. Si el registro **r12b** contiene la variable booleana **cond**, y **rax** la variable **valor**, la secuencia de instrucciones:

```
testb %r12b, %r12b
movq $13, %rax
cmovne $17, %rax
```

realiza la operación:

- a. valor = cond ? 13 : 17;
- b. valor = 17;
- c. valor = cond ? 17 : 13;
- d. valor = 13;

11. En una matriz declarada como “int a[n][n];” en lenguaje C...

- a. los n elementos de una columna se almacenan en memoria de manera contigua
- b. los n elementos de una fila se almacenan en memoria de manera contigua
- c. podría haber huecos de relleno al final de cada columna para alineamiento, dependiendo de n
- d. podría haber huecos de relleno al final de cada fila para alineamiento, dependiendo de n

12. ¿Cuáles de las siguientes señales son entradas a la unidad de control?

- a. El contenido del contador de programa
- b. Las señales de habilitación de buffers triestado entre registros y buses

c. El contenido del registro de instrucción

d. Las señales de control de la ALU

13. Una CPU con bus de direcciones de 64 bits y bus de datos de 32 bits tiene un registro de 64 bits conectado al bus de direcciones de la memoria. Probablemente se trata del registro

- a. IR
- b. MBR
- c. Acumulador
- d. MAR

14. En la secuencia de instrucciones siguiente, siendo el primer registro el destino, ¿cuántos riesgos se dan?

```
sub r2, r1, r3
or r8, r6, r1
```

- a. Un riesgo por dependencia de datos
- b. Un riesgo estructural
- c. Un riesgos por dependencia de datos y uno de control
- d. Ninguno

15. Un sistema no segmentado tarda 10 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) con un ciclo de reloj de 5 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se aproxima a:

- a. 2
- b. 5
- c. 10
- d. 20

16. ¿Cuál de las siguientes técnicas no se puede usar para determinar la causa de una interrupción?

- a. línea de reconocimiento INTA#
- b. interrupciones vectorizadas
- c. consulta de estado, o polling
- d. múltiples líneas de interrupción INT1#, INT2#... con un dispositivo en cada línea

17. Señale cuál de las siguientes opciones es una técnica para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:

- a. E/S por flanco
- b. E/S programada
- c. Acceso indirecto a memoria
- d. E/S por nivel

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- **Perfecto para largas noches de risas con amigos**



- 18.** ¿Cuál de las siguientes afirmaciones es **\*FALSA\***?
- La operación de lectura de una celda DRAM es destructiva
  - Las memorias DRAM son en general más lentas que las SRAM
  - Una celda DRAM no pierde la información al desconectar la alimentación**
  - Las memorias DRAM presentan generalmente una capacidad de almacenamiento mayor que las SRAM
- 
- 19.** La memoria DRAM:
- Se inventó en la década de los 2000
  - Necesita 6 transistores por cada celda
  - Se denomina dinámica porque su contenido puede alterarse, al contrario que la SRAM
  - Es más densa que la memoria SRAM**
- 
- 20.** Indique cuál es la dirección de la instrucción **mov** en el siguiente desensamblado, donde se ha borrado parte de la dirección
- ```
0804xxxx: 74 12          je 08048391
0804xxxx: b8 00 00 00 00 mov $0, %eax
```
- $08048391 + 12 = 08048403$
 - $08048391 - 12 = 08048379$
 - 0804837d
 - 0804837f**
-
- 21.** Dada la siguiente declaración en lenguaje C, una estructura de este tipo podría ocupar, bien sea en un sistema Linux IA32 o bien en uno x86-64, un total de...
- ```
struct a{
 int i;
 double d;
 char c;
 short s; };
```
- 18 B
  - 20 B
  - 22 B
  - 24 B**
- 
- 22.** Dado un camino de datos concreto, un posible formato de microprogramación se caracteriza como horizontal o vertical según tenga más o menos (señalar la respuesta **\*FALSA\***)
- codificación
  - solapamiento
  - microbifurcaciones**
- 
- d. longitud relativa de microinstrucción
- 23.** En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control codificadas de tal forma que pueda activarse sólo una o ninguna habrá de tener una anchura mínima de
- 4 bits
  - 5 bits**
  - 16 bits
  - 17 bits
- 
- 24.** Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señale la respuesta **\*FALSA\***)
- registros de acople (coste de la segmentación)
  - fragmentación desigual (duración desigual de etapas)
  - riesgos (*hazards*)
  - cola de instrucciones (precaptación)**
- 
- 25.** Un procesador de 1 GHz sin segmentación de cauce tarda 4 ns en ejecutar 4 instrucciones. ¿Cuánto tardaría en ejecutar 9 instrucciones una versión de dicho procesador con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones?
- 2 ns
  - 3 ns**
  - 4,5 ns
  - 9 ns
- 
- 26.** Respecto al salto retardado y al salto anulante, ¿cuál permite que se ejecute la siguiente instrucción, y cuál no?
- el retardado ejecuta la siguiente instrucción (con el correspondiente retraso), el anulante no la ejecuta (de hecho la anula)
  - el retardado la ejecuta sólo si se cumple la condición de salto, el anulante sólo si no se cumple
  - el retardado la ejecuta sólo si no se cumple la condición de salto, el anulante no la ejecuta nunca
  - el retardado la ejecuta siempre, el anulante la ejecuta sólo si se cumple la condición de salto**
-

**27.** Sobre la E/S mapeada en memoria podemos decir que:

- a. Usa el espacio común de direccionamiento para acceder a puertos de E/S
  - b. La CPU necesita el pin IO/M#
  - c. Dispone de instrucciones especiales de E/S
  - d. Todas las respuestas anteriores son falsas
- 

**28.** ¿Cuál de las siguientes tareas no es responsabilidad de un circuito de interfaz o controlador de periféricos sencillo?

- a. Adaptar el formato de las señales
  - b. Ajustar la temporización entre el procesador y los dispositivos de E/S
  - c. Recibir señales de control desde el procesador
  - d. Ejecutar el programa de transferencia de información entre el procesador y los dispositivos de E/S
- 

**29.** ¿Cuál es el ancho del bus de direcciones de un chip DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?

- a. 20
  - b. 16
  - c. 30
  - d. 15
- 

**30.** Sea un computador de 32 bits que dispone de una memoria cache de 512 KB y líneas de 64 bytes. ¿Cuántas líneas tiene la cache?

- a. 64
  - b. 1024
  - c. 8192
  - d. 65536
-

|                |               |
|----------------|---------------|
| <b>Nombre:</b> |               |
| <b>DNI:</b>    | <b>Grupo:</b> |

### Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |

1. El punto de entrada de un programa ensamblador en GNU/as Linux x86 se llama
- main
  - begin
  - \_start**
  - \_init

2. En un programa en ensamblador queremos crear espacio para una variable entera var inicializada a 1. La línea que hemos de escribir en la sección de datos es:
- .int var 1
  - var: .int 1**
  - .int: var 1
  - int var 1

3. En la práctica "media" se pide sumar una lista de 32 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. Un estudiante entrega la siguiente versión

```
$lista en EBX, longlista en ECX
suma:
```

```
 mov $0, %eax
 mov $0, %edx
 mov $0, %esi
```

```
bucle:
 add (%ebx,%esi,4), %eax
 jne nocarry
 inc %edx
```

```
nocarry:
 inc %esi
 cmp %esi,%ecx
 jne bucle
 ret
```

Esta función presenta una única diferencia frente a la solución recomendada en clase, relativa al salto condicional.

Esta función suma:

- produce siempre el resultado correcto
- fallaría con lista: .int 1,1,1,1, 1,1,1,1, ...
- fallaría con lista: .int 1,2,3,4, 1,2,3,4, ...
- no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**

4. En la misma práctica "media" un estudiante entrega la siguiente versión de suma sin signo:

```
main: .global main
...
call suma
...
mov $1, %eax
mov $0, %ebx
int $0x80
suma:
...
bucle:
...
nocarry:
 inc %esi
 cmp %esi,%ecx
 jne bucle
```

Notar que falta la instrucción ret final. Al desensamblar el código ejecutable se obtiene

```
08048445 <nocarry>:
08048445: 46 inc %esi
08048446: 39 f1 cmp %esi,%ecx
08048448: 75 f5 jne 804843f <bucle>
0804844a: 90 nop
0804844b: 90 nop
..... 90 nop
804844f: 90 nop
```



```

08048450 <__libc_csu_fini>:
08048450: 55 push %ebp
08048451: 89 e5 mov %esp,%ebp
08048453: 5d pop %ebp
08048454: c3 ret

```

- a. 2  
b. 3  
**c. 4**  
d. 5
- 

Este programa:

- está correctamente diseñado, la instrucción ret final es optativa, y no es concebible que quitar el ret cause algún error
  - produce un error "Segmentation fault" cuando empieza a acceder a memoria que no le corresponde (EIP)
  - funciona bien, pero si pusiéramos en el código fuente primero la definición de suma y luego la de main, el ejecutable terminaría accediendo a memoria que no le corresponde (ESP) y hará "Segmentation fault"**
  - no se puede marcar ninguna de las opciones anteriores
- 
- ¿Cuál de las siguientes sumas con signo produce desbordamiento con 32 bits?
    - 0xFFFFFFFF + 0xFFFFFFFF
    - 0x7FFFFFFF + 0xFFFFFFFF
    - 0x7FFFFFFF + 0x000000001**
    - 0xFFFFFFFF + 0x000000001

---

  - En la práctica “media” se pide sumar una lista de 32 enteros \*con\* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el menor valor positivo que repetido en toda la lista causaría overflow con 32bits?
    - 0x0400 0000**
    - 0x0800 0000
    - 0x4000 0000
    - 0x8000 0000

---

  - En la práctica “media” se pide sumar una lista de 32 enteros \*con\* signo de 32bits en una plataforma de 32bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista causaría overflow con 32bits?
    - 0xffff ffff
    - 0xfc00 0000
    - 0xfbff ffff**
    - 0xf000 0000

---

  - ¿Cuál es el popcorn (peso Hamming, nº de bits activados) del número 29?

- La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcorn4:

```

int popcorn4(unsigned* array, int len) {
 int val = 0;
 int i, j;
 unsigned x;
 int res = 0;
 for (i=0; i<len; i++) {
 x = array[i];
 val = 0;
 for (j=0; j<8; j++) {
 val += x & 0x01010101;
 x >>= 1;
 }
 val += (val>>16);
 val += (val>>8);
 res += val;
 }
 return (res & 0xFF);
}

```

Esta función presenta varias diferencias con la versión "oficial" recomendada en clase, incluyendo la "doble inicialización" de val y la acumulación y retorno de res. Esta popcorn4:

- produce siempre el resultado correcto
  - fallaría con array={0,1,2,3}
  - fallaría con array={1,2,4,8}
  - no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**
- 

- En la misma práctica "popcount" un estudiante entrega la siguiente versión de popcorn4:

```

int popcorn4(unsigned* array, int len) {
 int i, j;
 unsigned x;
 int result = 0;
 long val;
 for (i=0; i<len; i++) {
 x = array[i];
 val = 0;
 for (j=0; j<8*sizeof(int); j++) {
 val += x & 0x01010101;
 x >>= 1;
 }
 val += (val>>16);
 val += (val>>8);
 result += val & 0xFF;
 }
 return result;
}

```

# ¿BUSCAS LIBROS O EBOOKS SOBRE ENFERMERÍA?

AXON librería especializada en Ciencias de la Salud.

AXON  
axon.es



Encuentra todos los libros y eBooks para tu especialización, además del material complementario que necesites.

}

Esta función presenta varias diferencias con la versión "oficial" recomendada en clase, incluyendo el tipo de val y las condiciones del bucle for.

Esta función popcorn4:

- a. produce siempre el resultado correcto
  - b. fallaría con array={0,1,2,3}
  - c. fallaría con array={1,2,4,8}
  - d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
- 
11. Comparando los popcounts (pop(129) vs. pop(29)) y paridades (par(129) vs. par(29)) de los números 129 y 29, se verifica que
- a. pop(129) > pop(29)
  - b. par(129) > par(29)
  - c. pop(129) < par(29)
  - d. par(129) < pop(29)
- 

12. La práctica "parity" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity4:

```
int parity4(unsigned* array, int len){
 int val;
 int i;
 unsigned x;
 int result = 0;
 for (i=0; i<len; i++) {
 x = array[i];
 val = 0;
 asm("\n"
 "ini3: \n\t"
 "shr $0x1, %[x] \n\t"
 "adc $0x0, %[r] \n\t"
 "test %[x], %[x] \n\t"
 "jnz ini3 "
 : [r]"+r"(val)
 : [x]"r"(x)
);
 result += val & 0x1;
 }
 return result;
}
```

Esta función presenta una sentencia asm distinta de la versión "oficial" recomendada en clase. En concreto son distintas la etiqueta y las instrucciones adc/test.

Esta función parity4:

- a. produce siempre el resultado correcto
- b. fallaría con array={0,1,2,3}
- c. fallaría con array={1,2,4,8}

- d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
- 

13. En la misma práctica "parity" un estudiante entrega la siguiente versión de parity4:

```
int parity4(unsigned* array, int len){
 int i;
 unsigned x;
 int val, result = 0;
 for (i=0; i<len; i++) {
 x = array[i];
 val = 0;
 asm("\n"
 "ini4: \n\t "
 "xor %[x], %[y] \n\t"
 "shr $1, %[x] \n\t "
 "cmpl $0, %[x] \n\t "
 "jnz ini4 \n\t "
 :[y]"+r"(val)
 :[x]"r"(x)
);
 result += val & 0x1;
 }
 return result;
}
```

Esta función presenta dos diferencias con la versión "oficial" recomendada en clase, relativas a la instrucción cmp y al nombre [y] escogido para la restricción val. Esta parity4:

- a. produce siempre el resultado correcto
  - b. fallaría con array={0,1,2,3}
  - c. fallaría con array={1,2,4,8}
  - d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
- 

14. En la misma práctica "parity" un estudiante entrega la siguiente versión de parity6:

```
int parity6(int *array, int len){
 int i, res=0;
 unsigned x;
 for (i = 0; i < len; i++) {
 x = array[i];
 asm(
 "mov %[x], %%edx \n\t"
 "shr $16, %[x] \n\t"
 "xor %[x], %%edx \n\t"
 "xor %%dh, %%dl\n\t"
 "setpo %%dl \n\t"
 "movzx %%dl, %[x]\n\t"
 : [x]"+r"(x)
 :
 : "edx"
);
 res += (x & 0x1);
 }
 return res;
}
```

WUOLAH

}

Esta función presenta dos diferencias con la versión "oficial" recomendada en clase, relativas al tipo del array y a la máscara y paréntesis usados al acumular.

Esta función parity6:

- a. produce siempre el resultado correcto
  - b. fallaría con array={0,1,2,3}
  - c. fallaría con array={1,2,4,8}
  - d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
- 
15. En la práctica de la bomba, el primer ejercicio consistía en “saltarse” las “explosiones”, para lo cual se puede utilizar...
- a. objdump o gdb
  - b. **gdb o ddd**
  - c. ddd o hexedit
  - d. hexedit u objdump
- 
16. En la práctica de la bomba, el tercer ejercicio consistía en usar un editor hexadecimal para crear un ejecutable sin “explosiones”. Para saber qué contenidos del fichero hay que modificar, se puede utilizar... (marcar la opción \*FALSA\*)
- a. objdump
  - b. gdb
  - c. ddd
  - d. hexedit**
- 
17. Suponer una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponer que para la siguiente secuencia de direcciones enviadas a la cache: 0, 2, 4, 8, 16, 32, la tasa de acierto es 0.33. ¿Cuál es el tamaño de bloque de la cache?
- a. 4 bytes
  - b. 8 bytes**
  - c. 16 bytes
  - d. Ninguno de los anteriores
- 
18. Sea un computador de 32 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 2 vías. Dado el siguiente fragmento de código:

```
int v[262144];
for (i = 0; i < 262144; i += 8)
 v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?

- a. 0 (ningún fallo)
  - b. 1/2 (mitad aciertos, mitad fallos)**
  - c. 1/8 (un fallo por cada 8 accesos)
  - d. 1 (todo son fallos)
- 
19. En la práctica de la cache, el código de size.cc accede al vector saltando de 64 en 64. ¿Por qué?
- a. Porque cada elemento del vector ocupa 64 bytes
  - b. Para recorrer el vector más rápidamente
  - c. Porque el tamaño de cache L1 de todos los procesadores actuales es de 64KB
  - d. Para anular los aciertos por localidad espacial, esto es, que sólo pueda haber aciertos por localidad temporal**
- 
20. ¿En qué unidades se suelen medir las capacidades de almacenamiento de los niveles de cache L1, L2 y L3 de un microprocesador actual (2017-2018)?
- a. L1 en KB, L2 en KB o MB, L3 en MB.**
  - b. L1 en MB, L2 en GB, L3 en GB o TB.
  - c. L1 en MB, L2 en MB, L3 en GB.
  - d. L1 en KB, L2 en MB, L3 en GB.

**Nombre:**
**DNI:**
**Grupo:**

### Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

1. ¿Cuál es el valor mínimo (más negativo) que puede tomar un entero de 32 bits en complemento a dos? (el punto se usa como separador)

- a. -2.147.483.647
- b. **-2.147.483.648**
- c. -4.294.967.295
- d. -4.294.967.296

2. ¿Cómo se representa el valor -1 como entero con signo en 14 bits?

- a. 0xFFFF
- b. **0x3FFF**
- c. las respuestas anteriores no son válidas porque usan hexadecimal; habría que usar binario
- d. no se puede porque 14 no es múltiplo de 4

3. ¿Cuál de las siguientes no es una unidad de la arquitectura Von Neumann?

- a. Unidad central de proceso
- b. Memoria principal
- c. Sistema de entrada/salida
- d. **Núcleo del sistema operativo**

4. ¿Cuál de las siguientes afirmaciones es verdadera?

- a. La arquitectura Von Neumann en la que se basan los computadores tradicionales consiste en tener los datos separados de las instrucciones en memorias distintas.
- b. El registro de estado es un registro transparente al usuario, ya que éste no

puede utilizarlo en las instrucciones máquina.

- c. **El registro de instrucción es un registro transparente al usuario, ya que éste no puede utilizarlo en las instrucciones máquina.**
  - d. La unidad de control necesita como entrada el registro contador de programa, para saber cuál es la instrucción que debe ejecutar a continuación.
- 
5. ¿Qué es el lenguaje máquina?
- a. Conjunto de datos binarios que representan señales eléctricas internas de la unidad de control de un microprocesador.
  - b. Conjunto de sentencias en un lenguaje escrito que se utilizan para generar programas codificados en lenguaje ensamblador.
  - c. Conjunto formado por las siglas asignadas a las instrucciones del repertorio de instrucciones más un conjunto de directivas que facilitan la generación del código binario.
  - d. **Conjunto de instrucciones en formato binario que entiende un determinado procesador.**
- 
6. ¿Cuál de los siguientes elementos no forma parte de la Arquitectura del Repertorio de Instrucciones (ISA)?
- a. Descripción del espacio de direccionamiento de la memoria y de la E/S.

- b. Descripción de los campos de bits en los que están organizadas conceptualmente las microinstrucciones.
- c. Descripción de los registros de datos, registros de estado y control.
- d. Descripción de los tipos de datos sobre los que opera el lenguaje máquina.
- 
7. ¿Cuál de las siguientes definiciones de modos de direccionamiento es **\*incorrecta\***?
- a. Inmediato: el dato está codificado dentro de la propia instrucción, en uno de los campos en los que se divide el formato de instrucción.
- b. Registro: el dato se encuentra en un registro de propósito general.
- c. Directo: la dirección se calcula como la suma de un dato codificado en la propia instrucción y el contenido de un registro de propósito general.
- d. Indirecto: el dato está contenido en una posición de memoria que es apuntada por un registro de propósito general.
- 
8. Respecto a los registros enteros en arquitectura IA32 de 32bits (x86)
- a. Se puede acceder a 8, y en cada uno de esos 8 registros enteros, se puede acceder a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX) ó a los 8 LSBs (p.ej. AL)
- b. Se puede acceder a 8, y en cada uno de esos 8 registros enteros, se puede acceder a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX), a los 8 LSBs (p.ej. AL) o a los bits 8-15 (p.ej. AH)
- c. Se puede acceder a 8 de cada tamaño (32, 16, 8 bits), aunque no todos los registros tienen versión de 8 y 16 bits
- d. No hay distintos tamaños, son sólo registros de 32 bits, como corresponde a dicha arquitectura
- 
9. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):
- a. movw %dx, (%eax)
- b. movb \$0xFF, (%dl)
- c. movswl (%eax), %edx
- d. movzbl %dl, %eax
- 
10. ¿Qué modo de direccionamiento usa el operando fuente en la instrucción mov (%rcx), %al?
- a. Directo a memoria
- b. Indirecto a memoria a través de registro
- c. Registro
- d. Inmediato
- 
11. Si el contenido del registro %rax es 0x10 antes de ejecutar la instrucción shl \$0xc,%rax, ¿cuánto es su contenido tras ejecutarla?
- a. 0x10000
- b. 0x1000
- c. 0x4000
- d. 0x800
- 
12. En el fragmento de código
- ```
804854e: e8 3d 06 00 00 call 8048b90
8048553: 50           pushl %eax
```
- la instrucción call suma al contador de programa la cantidad:
- a. 0x0000063d
- b. 0x08048553
- c. 0x0804854e
- d. 0x50
-
13. ¿Cuál de los siguientes registros x86-64 es distinto del resto en convenio de uso? (salva-invocante/invocado)
- a. RBX
- b. RCX
- c. RSI
- d. R8
-
14. Respecto a requisitos de alineamiento de structs en gcc/IA32 x86 y x86-64, una de las siguientes afirmaciones es ***FALSA***
- a. en x86 Linux alinea double a 4x
- b. en x86 Linux alinea long double a 4x
- c. en x86-64 Linux alinea double a 8x
- d. en x86-64 Linux alinea float a 8x
-
15. Se definen las variables, unión y función C siguientes:
- ```
float f1;
unsigned u1=0x80000000;
float f2;
typedef union {
 float f;
 unsigned u;
} bit_float_t;
```

# ¿BUSCAS LIBROS O EBOOKS SOBRE ENFERMERÍA?

AXON librería especializada en Ciencias de la Salud.

AXON  
axon.es



Encuentra todos los libros y eBooks para tu especialización, además del material complementario que necesites.

```
float bit2float(unsigned u) {
 bit_float_t arg;
 arg.u = u;
 return arg.f;
}
```

¿Cuál afirmación es verdadera?

- a. Si asignamos f1=bit2float(u1); entonces f1== 2147483648.00
- b. Si asignamos f1=bit2float(u1); entonces f1== -0.0
- c. Si asignamos f2= (float)u1 ; entonces f2== 4294967296.00
- d. Si asignamos f2= float(u1); entonces f2== 0.0

16. Convertir un vector de 100 shorts de formato little endian a formato big endian consiste en:

- a. Intercambiar el elemento 0 del vector con el 99, el 1 con el 98, el 2 con el 97 y así sucesivamente.
- b. Intercambiar el elemento 0 del vector con el 1, el 1 con el 2, el 3 con el 4 y así sucesivamente.
- c. Cambiar el orden en memoria de los 4 bytes de cada elemento, es decir, en cada elemento intercambiar el byte 0 con el 3 y el 1 con el 2.
- d. Cambiar el orden en memoria de los 2 bytes de cada elemento, es decir, en cada elemento intercambiar el byte 0 con el 1.

17. Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señalar la respuesta \*FALSA\*)

- a. registros de acople (coste de segmentación)
- b. fragmentación desigual (duración de etapas)
- c. riesgos (hazards)
- d. cola de instrucciones (precaptación)

18. Un procesador de 1GHz tarda 4ns en realizar 4 instrucciones sin realizar segmentación de cauce. ¿Cuanto tardaría en realizar 9 instrucciones con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones?

- a. 2 ns
- b. 3 ns
- c. 4.5 ns
- d. 9 ns

19. ¿Qué es un controlador de E/S?

- a. Un circuito electrónico que implementa la memoria del computador.
- b. Un circuito impreso del tipo DIMM.
- c. Un circuito electrónico que puede guardar temporalmente datos enviados desde el procesador al periférico o viceversa.
- d. Un bus que permite interconectar distintos periféricos entre sí.

20. Respecto a la interfaz de E/S, ¿cuál de las siguientes afirmaciones es \*FALSA\*?

- a. Involucra tareas que se pueden implementar parte en hardware y parte en software.
- b. Permite configurar el funcionamiento del periférico en un momento determinado, y además conocer su estado.
- c. Puede guardar temporalmente en registros internos tanto datos generados por el periférico para ser enviados al procesador, como datos que son enviados desde el procesador al periférico.
- d. Una interfaz de entrada recibe los datos desde el procesador y los transforma y envía al periférico en formato digital.

21. ¿Cuál de las siguientes características corresponde a E/S mapeada en memoria?

- a. Determinadas zonas del espacio de direccionamiento del procesador se asignan por convenio a controladores de E/S.
- b. Un ejemplo de mecanismo de E/S mapeada en memoria es la instrucción IN de los procesadores Intel.
- c. Una misma dirección se usa alternativamente para E/S y para memoria en distintos momentos de ejecución de un programa.
- d. Un pin IO/M# del procesador permite distinguir si accedemos a E/S o a memoria.

22. ¿Cuál de las siguientes afirmaciones es \*FALSA\*?

- a. La consulta del estado del dispositivo por parte de la CPU se suele hacer con E/S programada (salvo con dispositivos que siempre están listos para transferir) y con E/S por IRQ (cuando se usa polling para determinar el origen de la IRQ).
- b. Se suele avisar a la CPU (mediante una IRQ) de que debe realizar alguna tarea, tanto en E/S por IRQ (obligatoriamente, la tarea es la transferencia) como en E/S por DMA (optativamente, el controlador DMA puede avisar de que acabó).

- c. Sólo E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S.
- d. Sólo E/S por DMA libera a la CPU de realizar la transferencia de los datos de E/S.

23. La instrucción máquina DI (Disable Interrupts), conocida como CLI (Clear Interrupt Flag) en x86, se utiliza para desactivar:

- a. Todas las interrupciones enmascarables
- b. Las interrupciones de inferior o igual prioridad a una dada
- c. Determinados niveles de interrupción de forma selectiva
- d. Las interrupciones software

24. ¿Cuál de los siguientes es un registro de un controlador de DMA?

- a. IR (Instruction Register)
- b. PC (Program Counter)
- c. SP (Stack Pointer)
- d. WC (Word Count)

25. El ancho de banda de memoria es:

- a. el número de bits que se pueden transferir entre ésta y la CPU en paralelo en una sola operación de lectura o escritura
- b. el número de bytes que se pueden leer/escribir por unidad de tiempo
- c. el tiempo que se tarda en transferir una palabra entre memoria y CPU
- d. el intervalo de frecuencias de reloj permitidas entre memoria y CPU

26. ¿Cuál de las siguientes afirmaciones sobre la memoria DRAM es \*incorrecta\*?

- a. El principio de funcionamiento de los circuitos electrónicos de la memoria DRAM consiste en cargar o descargar un transistor.
- b. Los bits de memoria se organizan dentro del circuito integrado en forma de matriz de celdas de bit, en la que se pueden diferenciar filas y columnas.
- c. Un transistor en cada celda permite o no permite circular la corriente eléctrica a través de él. Cuando el transistor no deja pasar la corriente, la información queda almacenada durante un tiempo en el condensador. Cuando el transistor deja pasar corriente, el condensador se carga o se descarga.
- d. Cada celda de memoria está compuesta por un transistor y un condensador y almacena un bit de información.

27. ¿Cuál de las siguientes afirmaciones sobre memorias es correcta?

- a. La memoria cache se construye con tecnología electrónica de tipo DRAM.
- b. La memoria principal se construye con tecnología electrónica de tipo SRAM.
- c. Los chips de memoria DRAM se conectan entre sí en un circuito impreso constituyendo lo que se denomina DIMM.
- d. Las memorias SRAM no son volátiles; es decir, cuando no están alimentadas eléctricamente siguen guardando toda la información.

28. ¿Cuál de los siguientes grupos de señales no se usa en un chip de memoria SRAM?

- a. Selección de filas RAS# y de columnas CAS#.
- b. Datos D<sub>n-1</sub>-D<sub>0</sub>.
- c. Direcciones A<sub>n-1</sub>-A<sub>0</sub>.
- d. Selección de chip CS# y habilitación de escritura WE#.

29. ¿Qué es el tiempo de refresco de memoria?

- a. La cantidad de datos transferidos por segundo entre dos niveles de la jerarquía de memoria.
- b. El tiempo que se tarda en recargar los condensadores que almacenan los bits de datos para que no se pierdan.
- c. El tiempo que transcurre entre la solicitud de una operación en un determinado nivel de la jerarquía de memoria (lectura o escritura) y la recepción de todos los datos solicitados.
- d. El tiempo que tiene que transcurrir entre sucesivas solicitudes de acceso a un determinado nivel de la jerarquía de memoria.

30. ¿Cuál de las siguientes políticas está \*menos\* relacionada con la jerarquía memoria?

- a. Política de escritura: determina cómo se actualiza el nivel de la memoria i+1 cuando se ejecutan instrucciones de almacenamiento en el nivel i.
- b. Política de reemplazo: qué bloque se tiene que sustituir (reemplazar) cuando se trae un bloque desde otro nivel.
- c. Política de planificación: en qué orden se ejecutarán los procesos pendientes.
- d. Política de colocación: dónde se almacena un bloque de datos dentro de la memoria.

|                |               |
|----------------|---------------|
| <b>Nombre:</b> |               |
| <b>DNI:</b>    | <b>Grupo:</b> |

### Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    | c  | d  |    |    |    |

1. Habiendo definido en código fuente ASM **longsal: .quad . -saludo** justo detrás de un string **saludo** que ocupaba 28 bytes, si se comparan los comandos gdb siguientes: **x/1xg &longsal** frente a **print (long) &longsal:**
  - a. ambos nos muestran la longitud del string (que es/vale/equivale a 28)
  - b. el primero (x) nos muestra el string, y el segundo (print) nos muestra otro valor distinto
  - c. el segundo (print) nos muestra el string, y el primero (x) nos muestra otro valor distinto
  - d. alguno o ambos contienen algún error gramatical (falta o sobra algún &, algún typecast (char\*) o (long), etc.)
2. En la práctica "media" se pide sumar una lista de 16 enteros SIN signo de 32 bits evitando acarreo. ¿Cuál es el menor valor que repetido en toda la lista causaría acarreo en 32 bits?
  - a. 0xFFFF FFFF
  - b. 0x7FFF FFFF
  - c. 0x1000 0000
  - d. 0x0FFF FFFF
3. En la práctica "media" se pide usar **adc** para sumar una lista de 16 enteros SIN signo de 32 bits en dos registros de 32 bits mediante extensión con ceros. Un estudiante entrega la siguiente versión:

```
...
resultado: .quad 0
...
main: .global main
 mov $lista, %rbx
 mov $16, %ecx
 call suma
```

```
mov %eax, resultado
código para printf ...
código para _exit ...
suma:
 mov $0, %eax
 mov $0, %rdx
bucle:
 adc (%rbx, %rdx, 4), %eax
 inc %rdx
 cmp %rdx, %rcx
 jnle bucle
 ret
```

Este programa no usa la variable **longlista**, guarda el resultado con una instrucción MOV, usa como índice RDX, no usa la instrucción ADD, y usa JNLE para el salto condicional.

Al empezar un programa CF no está activado. Esta versión de la suma SIN signo mediante extensión con ceros da resultado correcto:

- a. con lista: .int 0x10000000, ... (16 elementos)
  - b. con lista: .int 200000000, ... (16 elementos)
  - c. con ambos ejemplos
  - d. con ninguno de los dos ejemplos
4. En la práctica "media" se pide usar **cltd/cdq** para sumar una lista de 16 enteros CON signo de 32 bits en dos registros de 32 bits mediante extensión de signo. Un estudiante entrega la siguiente versión:

```
...
main: .global main
 mov $lista, %rbx
 mov longlista, %rcx
 call suma
 mov %eax, resultado
 mov %edx, resultado+4
```



```

movq $formato, %rdi
movq resultado,%rsi
movq resultado,%rdx
movl $0,%eax
call printf
...

```

El programa produce la siguiente salida con el test #01 (16 elementos con valor -1):

```

TEST01 -----
resultado = -16 (sng)
= 0x ffffffff00000000 (hex)
= 0x 00000010 9f816d80

```

Recordar que todo el texto aparecía tal cual literalmente en el formato (ignorar la errata sng) y los números llevaban especificación de formato (%18ld, %18lx, etc). De esta versión de la suma CON signo mediante extensión de signo se puede afirmar que:

- al inicio de main EAX vale 0 y R8 contiene un valor inferior a 0x7fffffff00000000
- al llamar a suma RBX contiene un valor inferior a 0x600000 y RCX vale 16
- al llamar a printf, ECX vale 16 y R8 contiene un valor superior a 0x80000000**
- tras volver de printf RAX contiene un valor superior a 60 y RDI superior a 0x600000
- En la práctica "media" se pide usar `cltq/cdq` y `cqto/cqo` para hallar la media y resto de una lista de 16 enteros CON signo de 32 bits usando registros de 64 bits. Un estudiante entrega la siguiente versión:

```

...
media: .double 0
resto: .double 0
formatoq:
.ascii "media = %11d resto = %11d\n"
.ascii "\t = 0x %08x \t = 0x %08x\n"
...
mov $lista, %rbx
mov longlista, %ecx
call sumaq

mov $formatoq, %rdi
mov media,%rsi
mov resto,%rdx
mov $0,%eax
call printf
...

sumaq:
push %rdx
push %rsi
mov $0, %rax
mov $0, %rsi
mov $0, %rdx
mov $0, %r8
bucleq:
mov (%rbx,%rsi,4), %eax
cdqe # EAX -> RAX

```

```

add %rax, %r8
inc %rsi
cmp %rsi,%rcx
jne bucleq
mov %r8, %rax
cqo # RAX -> RDX:RAX
idivq %rsi
mov %rdx, %r10
mov %rdx, resto
mov %rax, media
pop %rdx
pop %rsi
ret

```

Este programa es muy diferente a la versión "oficial" recomendada en clase. Notar los `push/pop`, los `mov $0` adicionales, `idiv %rsi` en lugar de `%rcx`, los `mov media/resto` al final de la subrutina en lugar de tras la llamada en `main`, y el tipo de ambas variables.

¿Qué media y resto imprime esta versión para el test #03? (16 elementos con valor 0x7fffffff)

- media = 2147483647 resto = 0**  
= 0x 7fffffff = 0x 00000000
- media = 16 resto = -16**  
= 0x 7fffffff = 0x 00000000
- media = 2147483647 resto = 0**  
= 0x 00000010 = 0x ffffff00
- media = 16 resto = -16**  
= 0x 00000010 = 0x ffffff00

- ¿Cuál expresión es cierta?

- `popcount(15) < popcount(51)`
- popcount( 2 ) == popcount(64)**
- `popcount( 7 ) > popcount(60)`
- `popcount(96) != popcount( 3 )`

- La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de `popcount1`:

```

int pc1(unsigned* array, size_t len) {
 size_t i,j;
 int res=0;
 unsigned x;
 for (i=0; i<len; i++) {
 x = array[i];
 for (j=0; j<8*sizeof(int); j++) {
 x >>= 1;
 unsigned bit = x & 0x1;
 res+=bit;
 }
 }
 return res;
}

```

Esta función se diferencia de la versión "oficial" recomendada en clase en el cuerpo del bucle interno. Esta función `popcount1`:

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- **Perfecto para largas noches de risas con amigos**



- a. produce siempre el resultado correcto  
 b. fallaría con array={0,1,2,3}  
 c. fallaría con array={1,2,4,8}  
 d. **no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**

8. En la misma práctica "popcount" un estudiante entrega la siguiente versión de popcount2:

```
int pcount2(int* array, size_t len) {
 size_t i;
 int res=0;
 unsigned x;
 unsigned bit;
 for (i=0; i<len; i++) {
 x = array[i];
 while(x) {
 bit += x & 0x1;
 x >>= 1;
 res = res + bit;
 }
 }
 return res;
}
```

Esta función se diferencia de la versión "oficial" recomendada en clase en el tipo del **array**, la variable **bit** y el cuerpo del bucle interno. Esta función popcount1:

- a. produce siempre el resultado correcto  
 b. fallaría con array={0,1,2,3}  
 c. fallaría con array={1,2,4,8}  
 d. **no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos**

9. En la misma práctica "popcount" un estudiante entrega la siguiente versión de popcount4:

```
int pc4(unsigned* array, size_t len) {
 size_t i;
 int res=0;
 unsigned x;
 for (i=0; i<len; i++) {
 x = array[i];
 asm("\n\t"
 "clc \n"
 "ini4: \n\t"
 "adc $0, %[r] \n\t"
 "test %[x],%[x] \n\t"
 "shr %[x] \n\t"
 "jne ini4 \n\t"
 "adc $0, %[r] \n\t"
 : [r]"+r" (res)
 : [x] "r" (x));
 }
 return res;
}
```

Esta función se diferencia de la versión "oficial" en que tiene una instrucción ensamblador adicional. Este popcount4:

- a. **produce siempre el resultado correcto**  
 b. fallaría con array={0,1,2,3}

- c. fallaría con array={1,2,4,8}  
 d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

10. En la misma práctica "popcount" un estudiante entrega la siguiente versión de popcount4:

```
int pc4(unsigned* array, size_t len) {
 size_t i;
 int res=0;
 unsigned x;
 for (i=0; i<len; i++) {
 x = array[i];
 asm("\n\t"
 "clc \n"
 "ini4: \n\t"
 "adc $0, %[r] \n\t"
 "fin4: \n\t"
 "shr %[x] \n\t"
 "jne ini3 \n\t"
 : [r]"+r" (res)
 : [x] "r" (x));
 }
 return res;
}
```

Esta función es muy diferente a la versión "oficial". Notar el salto condicional a "ini3" en la función popcount3 (en donde sí se hizo bien el **ini3:/shr/adc/test/jne** recomendado). Esta función popcount4:

- a. produce siempre el resultado correcto  
 b. fallaría con array={0,1,2,3}  
 c. fallaría con array={1,2,4,8}  
 d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

11. En la práctica de la bomba, el primer ejercicio consistía en saltarse las explosiones, para lo cual se puede utilizar... (marcar opción **falsa**)

- a. objdump  
 b. gdb  
 c. ddd  
 d. eclipse

12. ¿Para qué se utiliza la función gettimeofday() en la práctica de la "bomba digital"?

- a. Para cronometrar y poder comparar lo que tardan las distintas versiones del programa  
 b. Para imprimir la hora en la pantalla  
 c. Para cifrar la clave en función de la hora actual  
 d. **Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la clave**

13. ¿Para qué se utiliza la función scanf() en la práctica de la "bomba digital"?

- a. Para escanear el fichero ejecutable "bomba" y asegurarse de que no contenga virus

- b. Para leer la contraseña (clave alfanumérica)
- c. Para leer el PIN (clave numérica)
- d. Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la clave
- 
14. Respecto a las bombas estudiadas en la práctica "bomba digital", ¿en cuál de los siguientes tipos de bomba sería más **difícil** descubrir la contraseña? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario por teclado. Por "cifrar" podemos entender la cifra del César, por ejemplo.
- 1 string del fuente se cifra, se invierte y se compara con el string del usuario
  - el string del usuario se cifra y se compara con 1 string del fuente**
  - 2 strings del fuente se invierten, se concatenan, se cifra el resultado, y se compara con el string del usuario
  - el string del usuario se concatena con 1 string del fuente, luego se invierte 1 string del fuente, se cifra y se compara con el concatenado
- 
15. En una bomba como las estudiadas en prácticas, del tipo...
- ```

0x40080f <main+180> lea 0xc(%rsp),%rsi
0x400814 <main+185> lea 0x1dd(%rip),%rdi
                                # 0x4009f8
0x40081b <main+192> mov    $0x0,%eax
0x400820 <main+197> call  0x400620<scanf>
0x400825 <main+202> mov    %eax,%ebx
0x400827 <main+204> test   %eax,%eax
0x400829 <main+206> jne   0x40083c <m+225>
0x40082b <main+208> lea   0x1c9(%rip),%rdi
                                # 0x4009fb
0x400832 <main+215> mov    $0x0,%eax
0x400837 <main+220> call  0x400620<scanf>
0x40083c <main+225> cmp    $0x1,%ebx
0x40083f <main+228> jne   0x4007f9 <m+158>
0x400841 <main+230> mov   0x200819(%rip),
                           %eax # 0x601060
0x400847 <main+236> cmp    %eax,0xc(%rsp)
0x40084b <main+240> je    0x400852 <m+247>
0x40084d <main+242> call  0x400727 <boom>
0x400852 <main+247> lea   0x10(%rsp),%rdi

```
- ...el código numérico (pin) es...
- el entero 0x601060
 - el entero cuya dirección está almacenada en la posición de memoria 0x4009f8
 - el entero almacenado a partir de la posición de memoria 0x4009fb
 - el entero almacenado a partir de la posición de memoria 0x200819+0x400847**
-
16. La función **setup()** de Arduino es llamada:
- Al principio de cada iteración de la función **loop()**
-
- b. Cuando se sube desde el **kit entorno** de desarrollo o se pulsa el botón de reset, pero no cuando se conecta la alimentación
- c. Cuando se conecta la alimentación a la placa, se pulsa el botón de reset, o se sube desde el **kit entorno** de desarrollo
- d. Cuando se conecta la alimentación a la placa, se pulsa el botón de reset, pero no cuando se sube desde el **kit entorno** de desarrollo
-
17. ¿Qué sentencia usamos en el programa **blink** (led intermitente) para encender el led integrado en la placa Elegoo Mega2560?
- digitalWrite (LED_BUILTIN, HIGH);**
 - pinMode (LED_BUILTIN, OUTPUT);
 - analogWrite (LED_BUILTIN, HIGH);
 - pulseIn (LED_BUILTIN, HIGH);
-
18. Sobre el resultado devuelto por la función de Arduino **map(sensorValue, sensorLow, sensorHigh, 50, 4000)**; usada en el programa del Theremín de luz:
- Si **sensorValue** vale 50, devuelve **sensorLow**
 - Si **sensorValue** vale 50, devuelve **sensorHigh**
 - Si **sensorValue** vale **25202025**, devuelve la mitad entre **sensorLow** y **sensorHigh**
 - Si **sensorValue** vale la mitad entre **sensorLow** y **sensorHigh**, devuelve **25202025****
-
19. En el programa **line.cc** de la práctica de cache, si para cada tamaño de línea (**line**) recorremos una única vez el vector, la gráfica resultante es decreciente porque:
- hay un mayor historial de accesos y es más probable que un nuevo acceso sea un acierto
 - cada vez los tamaños de línea escogidos van decreciendo y se tarda menos en leerlos
 - cada vez los tamaños de línea escogidos van decreciendo y hay menor localidad espacial
 - el vector se indexa con la variable de control del bucle, con un incremento o paso de **line****
-
20. En la práctica de la cache, en **size.cc** se accede al vector saltando de 64 en 64. ¿Por qué?
- Para recorrer el vector más rápidamente
 - Porque con un salto menor que 64 habría aciertos por localidad espacial y haría menos clara la gráfica**
 - Porque cada elemento del vector ocupa 64 B
 - Para evitar aciertos por localidad temporal y que sólo haya aciertos por localidad espacial

Nombre:
DNI:
Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Cuál es el complemento a 2 del número binario 1110 1101 1000?

- a. 0001 0010 0110
- b. 0001 0010 0101
- c. 0001 0010 0111
- d. 0001 0010 1000**

d. 0xfffffffffffffde1

5. Para comprobar si el contenido del registro RDX es 0 (y posiblemente saltar a continuación usando la instrucción je), el compilador gcc genera:

- a. cmpq %rdx, %rdx
- b. testq %rdx
- c. testq %rdx, %rdx**
- d. cmpq %rdx

6. Sabiendo que las instrucciones de salto condicional codifican la dirección de salto con direccionamiento relativo a contador de programa (de 8 o 32 bits con signo), indicar cuál es la dirección de salto de la instrucción je en el siguiente desensamblado, donde se ha tachado precisamente dicha dirección.

```
40042f: 74 f4      je xxxxxxxx
400431: 5d          pop %rbp
```

- a. 400431
- b. 400525
- c. 400425**
- d. 40043d

7. Para traducir una asignación condicional (**a = b ? c : d ;**) de lenguaje C a lenguaje ensamblador, gcc puede que utilice...

- a. Un salto incondicional, según la condición expresada en el código C, y otro salto incondicional
- b. Un salto condicional, según la condición opuesta a la del código C, y otro salto condicional

c. Una instrucción de movimiento condicional, pero sólo si el procesador es Pentium Pro/II o superior

d. Una instrucción de movimiento incondicional, pero sólo si el S.O. es de 64bits

8. La instrucción **cmove %rdx,%rax**

- a. copia el byte bajo de rdx en el byte bajo de rax
- b. copia en rax el byte de memoria apuntado por la dirección contenida en rdx
- c. copia en rax el contenido de rdx si rax es menor que rdx

d. copia en rax el contenido de rdx si CF= 1

9. Uno de los puntos clave de la traducción que gcc hace de una construcción switch-case de lenguaje C a lenguaje ensamblador es...

- a. el salto condicional hacia atrás
- b. el salto relativo a contador de programa
- c. el salto directo

d. el salto indirecto

10. El procesador utiliza el puntero de pila...

- a. En las instrucciones de llamadas y retornos de subrutinas
- b. En todo tipo de instrucciones de saltos, incluyendo llamadas y retornos a subrutinas
- c. En todas las instrucciones que tengan al menos dos accesos a memoria
- d. En todas las instrucciones

11. ¿Cuál de las siguientes instrucciones situada al principio de una función se utilizará probablemente para crear espacio en la pila para variables locales sin inicializar?

- a. sub \$0x30, %rsp
- b. add \$0x30, %rsp
- c. sub \$0x30, %rbp
- d. add \$0x30, %rbp

12. En la convención de llamada SystemV AMD64 seguida por gcc Linux/x86-64...

- a. RAX es un registro salva-invocante, por eso en cualquier función hay que salvarlo antes de modificarlo
- b. R10 es un registro salva-invocante, por eso si es necesario hay que salvarlo antes de llamar a función

c. R11 es un registro salva-invocado, por eso en cualquier función hay que salvarlo antes de modificarlo

d. RBP es un registro salva-invocado, por eso si es necesario hay que salvarlo antes de llamar a función

13. Un procedimiento llamado por una instrucción call debe guardar y restaurar los registros siguientes siempre que los altere:

- a. %rsi, %rdi
- b. %rax, %rbx, %rcx, %rdx
- c. %rax, %rdx, %rcx

d. %rbx, %rbp

14. Dada una función que devuelve la suma de 8 enteros en x86-64, ¿cuál de las siguientes instrucciones suma el 7º argumento?

- a. add -0x8(%rsp), %eax
- b. add 0x8(%rsp), %eax
- c. add -0x4(%rsp), %eax
- d. add 0x4(%rsp), %eax

15. En el fragmento de programa siguiente:

66b: e8 8a ff ff ff callq 5fa <f>
670: 48 83 c4 10 add \$0x10,%rsp

¿Cuál es el valor que introduce en la pila la instrucción callq?

- a. 0x670
- b. 0xffffffff8a
- c. 0x66b
- d. 0x5fa

16. En el fragmento de programa siguiente:

66b: e8 8a ff ff ff callq 5fa <f>
670: 48 83 c4 10 add \$0x10,%rsp

la instrucción callq suma al contador de programa la cantidad:

- a. -0x76
- b. 0x5fa
- c. 0xffffffff8a
- d. 0x76

17. Suponga la siguiente llamada a una función f de 4 argumentos:

mov \$0x1, %ecx
mov \$0x2, %edx
mov \$0x3, %esi
mov \$0x4, %edi
callq 5fa <f>

El primer parámetro de llamada a la función:

WUOLAH

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- **Perfecto para largas noches de risas con amigos**



- a. Es el valor inmediato 1
 b. Es el contenido de la dirección de memoria 0x1
c. Es el valor inmediato 4
 d. Es el contenido de la dirección de memoria 0x4

- 18.** Habiendo declarado **int array={0,1,2,3};** y **long long *ptr=array;** ¿cuánto vale **ptr[1]**?
- a. 0x0000 0001 0002 0003
 b. 0x0000 0001 0000 0000
 c. 0x0003 0002 0001 0000
d. 0x0000 0003 0000 0002

- 19.** Una función C llamada **get_el(...)** genera el siguiente código ensamblador.

```
leaq    (%rdi,%rdi,4), %rax
addq    %rax, %rsi
movl    arr(%rsi,4), %eax
ret
```

Se puede adivinar que:

- a. arr es un array multi-nivel (punteros a enteros) de cuatro filas
 b. arr es un array multi-nivel pero no se pueden adivinar las dimensiones
 c. arr es un array bidimensional de enteros, no se pueden adivinar dimensiones
d. arr es un array bidimensional de enteros, con cinco columnas

- 20.** Las microoperaciones de la fase de captación de una instrucción:

- a. Son comunes para todas las instrucciones**
 b. Dependen del código de operación de la instrucción que se encuentra en el registro de instrucción
 c. Dependen de los indicadores de estado y del código de operación de la instrucción que se encuentra en el registro de instrucción
 d. Dependen del valor del contador de programa

- 21.** Para el procesador con unidad de control microprogramada estudiado en clase, Tanenbaum propone codificar los 16 registros y añadir una señal “PERC” para habilitar la carga desde el bus C (recordar que era un diseño típico con 3 buses) y así no perder expresividad/parallelismo. El ahorro de bits en cada microinstrucción debido a esta técnica es de

- a. 40 bits
 b. 39 bits
c. 35 bits
 d. 29 bits

- 22.** Un procesador está segmentado en k etapas. Cada una de ellas consume un tiempo t. La aceleración ideal (si no hay riesgos) al ejecutar 5 instrucciones respecto a un procesador no segmentado será:
- a. **5k / (4+k)**
 b. **(4+k) / 5t**
 c. **4k / (5+k)**
 d. **(5+k) / 4t**

- 23.** ¿Cuál de las siguientes afirmaciones sobre la E/S programada con consulta de estado es cierta?

- a. Si se emplea E/S programada puede hacerse con consulta de estado o sin consulta de estado**
 b. Un programa que realice salida programada con consulta de estado no ejecutará ninguna instrucción de entrada o carga
 c. Sólo la E/S por DMA libera a la CPU de realizar la consulta de estado del dispositivo de E/S
 d. La escritura de un led requiere consulta de estado

- 24.** En un sistema de interrupciones vectorizado y en daisy-chain, ¿cuál de las siguientes afirmaciones es cierta?

- a. El procesador informa de un ciclo de reconocimiento de interrupción con la señal de reconocimiento de interrupción (INTA) y la identificación de los dispositivos se realiza por consulta de estado
b. La gestión de prioridades queda establecida por el orden en que los dispositivos reciben la señal INTA y el dispositivo se identifica por un dato que deposita en el bus
 c. La gestión de prioridades queda establecida por el orden en que los dispositivos reciben la señal INTA y la identificación de los dispositivos se realiza leyendo sus registros de estado
 d. El daisy-chain asigna a todos los dispositivos la misma prioridad y la identificación de los dispositivos se realiza leyendo sus registros de estado

25. ¿Cuál de las siguientes características es menos probable que pueda programarse en un canal DMA?

- a. dos direcciones (origen y destino)
 - b. dos tamaños (copia origen y copia destino)**
 - c. cuál de las dos direcciones es de E/S (si alguna lo es) en lugar de Memoria
 - d. si se desea producir una IRQ al terminar
-

26. En un computador con una jerarquía de memoria de dos niveles se observa experimentalmente que el tiempo medio de acceso a la memoria es de 300 ns cuando en realidad el tiempo medio de acceso al primer nivel es de 6 ns. Sabiendo que el tiempo de acceso al segundo nivel es de 3 microsegundos, ¿cuál sería aproximadamente el porcentaje de fallos en los accesos al primer nivel?

- a. 90%
 - b. 1%
 - c. 10%**
 - d. 99%
-

27. El orden de magnitud del tiempo de acceso a la memoria DRAM de un computador es de:

- a. Picosegundos
 - b. Nanosegundos**
 - c. Microsegundos
 - d. Milisegundos
-

28. Una memoria estática tiene un bus de datos de 32 bits y su bus de direcciones es de 20 bits, ¿cuál es su capacidad?

- a. 4 MBytes**
 - b. 1 MByte
 - c. 32 MBytes
 - d. 80 GBytes
-

29. En una cache asociativa por conjuntos de 2^v vías con 2^b líneas (marcos de bloque) de 2^w palabras, el gestor de memoria **no** considera como campo (conjunto de bits contiguos con significado o relevancia) los siguientes bits:

- a. últimos w bits (0...w-1) (los menos significativos)
 - b. bits w...w+c-1 (con c=b-v)
 - c. bits w...w+c-1 (siendo $2^c=n$ conjuntos)
 - d. bits b...b+c-1 (siendo $2^c=n$ conjuntos)**
-

30. Para obtener una única velocidad comparativa final, el benchmark SPEC CPU combina las ganancias en velocidad de ejecución de una serie de tests, respecto a un ordenador de referencia, usando...

- a. la mediana
 - b. la media aritmética
 - c. la media geométrica**
 - d. la moda
-

Nombre:
DNI:
Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros en doble precisión. Alguna de las siguientes técnicas no está relacionada con la aritmética en doble precisión:

- a. acumulación de desbordamientos (OF, overflow flag)
- b. acumulación de acarreos (CF, carry flag)
- c. extensión con ceros
- d. extensión de signo

2. En la misma práctica "media", el programa esqueleto ofrecido (suma.s) no es válido, en concreto... (marcar la opción FALSA)

- a. no está preparado para sumar más de 9 elementos
- b. no hace extensión con ceros de los elementos
- c. no hace extensión de signo de los elementos
- d. no consulta ni el flag de acarreo CF ni el de overflow OF

3. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de `popcount5`:

```
int popcount5(int* array, size_t len){
    size_t i,j;
    int x, val, result=0;
    for (i=0; i<len; i++){
        x = array[i];
        val = 0;
        for (j=0; j<8; j++){
            val += x & 0x01010101;
            x >>= 1;
        }
        val += (val >> 16);
        val += (val >> 8);
        result+= val & 0xFF;
    }
}
```

```
    }
    return result;
}
```

Esta función se diferencia de la versión "oficial" en los tipos de array y x.

Esta función `popcount5`:

- a. produce siempre el resultado correcto
- b. fallaría con array={0,1,2,3}
- c. fallaría con array={0,-1,-2,-3}
- d. no es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

4. Otro estudiante entrega la siguiente versión de `popcount5`:

```
int pcnt5(unsigned* array, size_t len) {
    size_t i,j;
    unsigned x;
    int val, result=0;

    for (i=0; i<len; i++) {
        x = array[i];
        val = 0;
        for (j=0; j<=8; j++) {
            val += x & 0x01010101;
            x >>= 1;
        }
        val += (val >> 16);
        val += (val >> 8);
        result+= val & 0xFF;
    }
    return result;
}
```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en las condiciones del for interno.

Esta función `popcount5` fallaría:

- a. con `array={1, 16, 256, 4096}`
 - b. con `array={1, 32, 1024, 32768}`
 - c. con ambos ejemplos
 - d. con ninguno de los dos ejemplos
-

5. En una bomba como las estudiadas en prácticas, del tipo...

```
4006fa: lea    0x10(%rsp),%rbx
4006ff: mov    0x20096a(%rip),%rdx
400706: mov    $0x64,%esi
40070b: mov    %rbx,%rdi
40070e: callq  400570 <fgets@plt>
400713: mov    $0xd,%edx
400718: lea    0x200939(%rip),%rsi
40071f: mov    %rbx,%rdi
400722: callq  400560 <strncmp@plt>
400727: test   %eax,%eax
400729: je     400730 <main+0x51>
40072b: callq  400697 <boom>
400730: lea    0x1b5(%rip),%rsi
```

...la contraseña (alfanumérica) es...

- a. el string almacenado a partir de `0x10(%rsp)`
 - b. el string alm. a partir de `0x20096a+0x4006ff`
 - c. el string almacenado a partir de `0x601058`
 - d. el string alm. a partir de `0x200939+0x400718`
-

6. En una bomba como las estudiadas en prácticas, del tipo...

```
400746: lea    0xc(%rsp),%rsi
40074b: lea    0x1ae(%rip),%rdi
400752: mov    $0x0,%eax
400757: callq  400590 <_scanf@plt>
40075c: mov    0x2008ee(%rip),%eax
400762: cmp    %eax,0xc(%rsp)
400766: je     40076d <main+0x8e>
400768: callq  400697 <boom>
40076d: callq  4006bb <defused>
```

...el código numérico (pin) es...

- a. el entero almacenado a partir de `0xc(%rsp)`
 - b. el entero alm. a partir de `0x1ae+0x40074b`
 - c. el entero almacenado a partir de `0x6010560`
 - d. el entero alm. a partir de `0x2008ee+0x400762`
-

7. La función `setup()` de Arduino es llamada:

- a. Al principio de cada iteración de la función `loop()`
- b. Cuando se sube desde el entorno de desarrollo o se pulsa el botón de reset, pero no cuando se conecta la alimentación

c. Cuando se conecta la alimentación a la placa, se pulsa el botón de reset, o se sube desde el entorno de desarrollo

d. Cuando se conecta la alimentación a la placa, se pulsa el botón de reset, pero no cuando se sube desde el entorno de desarrollo

8. Suponga una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponga que para la siguiente secuencia de direcciones enviadas a la cache: 0, 1, 2, 4, 8, 16, 32, 64, la tasa de acierto es 0,25. ¿Cuál es el tamaño de línea de la cache?

- a. 4 bytes
 - b. 8 bytes
 - c. 16 bytes
 - d. Ninguno de los anteriores
-

9. En el programa `size.cc` de la práctica "cache", se accede al vector saltando...

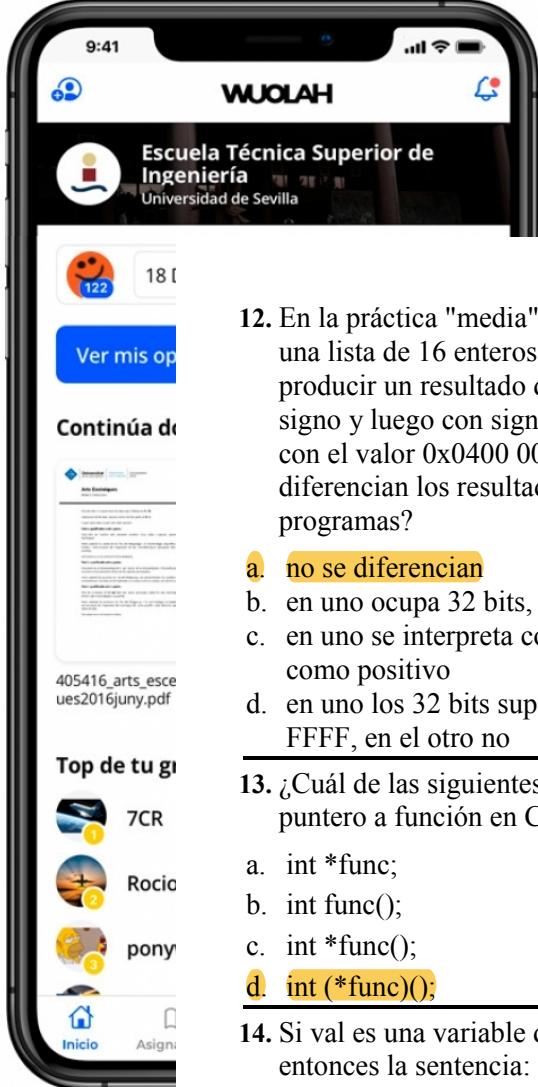
- a. de byte en byte
 - b. de 64 en 64 bytes
 - c. de 1 KB en 1 KB
 - d. de line en line bytes, donde line barre los valores desde 1 hasta 1K en un bucle for
-

10. En la práctica de cache hemos hecho una gráfica con el código `size.cc` ¿Qué forma tiene la gráfica que se debe obtener?

- a. Forma de U (o V), con un tramo descendente y otro ascendente
 - b. Forma de /, una gráfica siempre creciente y sin escalones
 - c. Forma de media U seguida de \, es decir, un tramo descendente suave, un pequeño tramo horizontal, y un tramo descendente lineal
 - d. Una escalera con varios tramos horizontales
-

11. La línea de código ensamblador: **mov \$msg, %rsi**

- a. Copia en rsi los primeros 64 bits de memoria desde la posición apuntada por la etiqueta msg.
 - b. Copia en rsi todo el contenido de la cadena apuntada por msg.
 - c. Copia en rsi la dirección de memoria de 64 bits almacenada en memoria a partir de la posición indicada por la etiqueta msg.
 - d. Copia en rsi los 64 bits de la dirección msg.
-



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

12. En la práctica "media" se programa la suma de una lista de 16 enteros de 4 bytes para producir un resultado de 8 bytes, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x0400 0000, ¿en qué se diferencian los resultados de ambos programas?
- no se diferencian**
 - en uno ocupa 32 bits, en otro 64 bits
 - en uno se interpreta como negativo, en otro como positivo
 - en uno los 32 bits superiores son 0xFFFF FFFF, en el otro no
-
13. ¿Cuál de las siguientes líneas declara un puntero a función en C?
- int *func;
 - int func();
 - int *func();
 - int (*func)();**
-
14. Si val es una variable de tipo unsigned long, entonces la sentencia: **val += (val >> 32);**
- Pone siempre val a 0.
 - Deja siempre val al mismo valor que tuviera antes de la sentencia.
 - Su traducción incluye una instrucción shr seguida de una suma.**
 - Su traducción incluye una instrucción sar seguida de una suma.
-
15. ¿Para qué se utiliza la función gettimeofday en la práctica de la "bomba digital"?
- Para cronometrar y poder comparar las duraciones de las distintas soluciones del programa.
 - Para imprimir la hora en la pantalla.
 - Para cifrar la contraseña en función de la hora actual.
 - Para lanzar un error cuando el usuario tarde demasiado tiempo en introducir la contraseña o el PIN.**
-
16. En la práctica de la bomba, el primer ejercicio consiste en ir saltando las “explosiones” mientras se depura el código, para lo cual se puede utilizar...
- objdump o gdb
 - gdb o ddd**
 - ddd o hexedit

- d. hexedit u objdump
-

17. En la placa del kit de Arduino, las patillas de tierra vienen etiquetadas con la leyenda:

- A0
 - 5V
 - GND**
 - 3.3V
-

18. Las resistencias utilizadas en la práctica de Arduino

- Son de color azul claro y tienen 5 bandas de color: las 3 primeras indican un valor, la 4^a banda es un multiplicador y la 5^a banda es la tolerancia
 - Son de color beige y tienen 4 bandas de color: las 2 primeras indican un valor, la 3^a banda es un multiplicador y la 4^a banda es la tolerancia
 - Tienen polaridad y el cátodo (polo negativo) es el extremo de la banda de color con una separación mayor respecto a las otras.
 - Tienen polaridad y el ánodo (polo positivo) es el extremo de la banda de color con una separación mayor respecto a las otras.
-

19. ¿Cuál de las siguientes afirmaciones sobre las caches es FALSA?

- Casi ningún procesador actual tiene memoria cache L2.**
 - Las direcciones a las que accede un programa no son completamente aleatorias, sino que se rigen por ciertos patrones de localidad.
 - Un procesador actual tiene varias caches de nivel 1.
 - La cache de nivel 3 no contiene toda la memoria que maneja el programa.
-

20. En el programa line.cc, si para cada tamaño de línea (line) recorremos una única vez el vector, la gráfica resultante es decreciente porque:

- Cada vez que line aumenta al doble, el número de aciertos por localidad temporal aumenta, porque ya habíamos accedido a cada posición i del vector cuando lo recorrimos en el punto anterior del eje X.
- Cada vez que line aumenta al doble, el número de aciertos por localidad espacial aumenta, porque ya habíamos accedido a cada posición i-1 del vector cuando lo recorrimos en el punto anterior del eje X.

- c. Cada vez que line aumenta al doble, se accede con éxito a más posiciones del vector en niveles de la jerarquía de memoria más rápidos.
 - d. Cada vez que line aumenta al doble, realizamos la mitad de accesos al vector que para el valor anterior.
-

Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. En el direccionamiento indirecto a través de registro, la dirección efectiva...
 - a. se encuentra en una dirección de memoria
 - b. **se encuentra en un registro general del procesador**
 - c. se encuentra en el registro de instrucción
 - d. se calcula como la suma del contenido de dos registros

2. Si N es el número de instrucciones máquina de un programa, F es la frecuencia de reloj, y C el número promedio de ciclos por instrucción, el tiempo de ejecución del programa será:
 - a. $N \cdot F/C$
 - b. **$N \cdot C/F$**
 - c. $N \cdot F \cdot C$
 - d. $N/(C \cdot F)$

3. ¿Cuál/es de los siguientes tipos de datos tienen distinto tamaño en arquitecturas x86 y x86-64 (Intel 32 bits y 64 bits)?
 - a. long
 - b. char *
 - c. ninguno de los dos
 - d. **ambos**

4. La secuencia de instrucciones


```
leaq (%rdi, %rdi, 2), %rax
salq $3, %rax
```

produce el efecto de...
 - a. leer en RAX a partir de la posición $2 * \text{RDI} + 2$ y multiplicar dicho contenido por 8

- b. calcular $\text{RAX} = (2 * \text{RDI} + 2) * 3 = 6 * (\text{RDI} + 1)$
- c. **calcular $\text{RAX} = 24 * \text{RDI}$**
- d. calcular $\text{RAX} = 9 * \text{RDI}$

-
5. ¿Qué valor contendrá el registro RDX tras ejecutar las dos instrucciones siguientes?

```
movq $-1, %rdx
movb $1, %dl
```

- a. 0xFFFF FFFF FFFF FFF1
- b. **0xFFFF FFFF FFFF FF01**
- c. 0xFFFF FFFF FFFF 0001
- d. 0x0000 0000 0000 0001

-
6. En la secuencia de programa siguiente:

```
400544: e8 07 00 00 00 callq 400550 <f>
400549: 48 89 03          mov %rax, (%rbx)
```

¿cuál es el valor que introduce en la pila la instrucción call?

- a. 0x400544
- b. **0x400549**
- c. 0x40054b
- d. 0x400550

-
7. Una función C declarada como **int get_var_digit(size_t index, size_t digit)** genera como código ensamblador

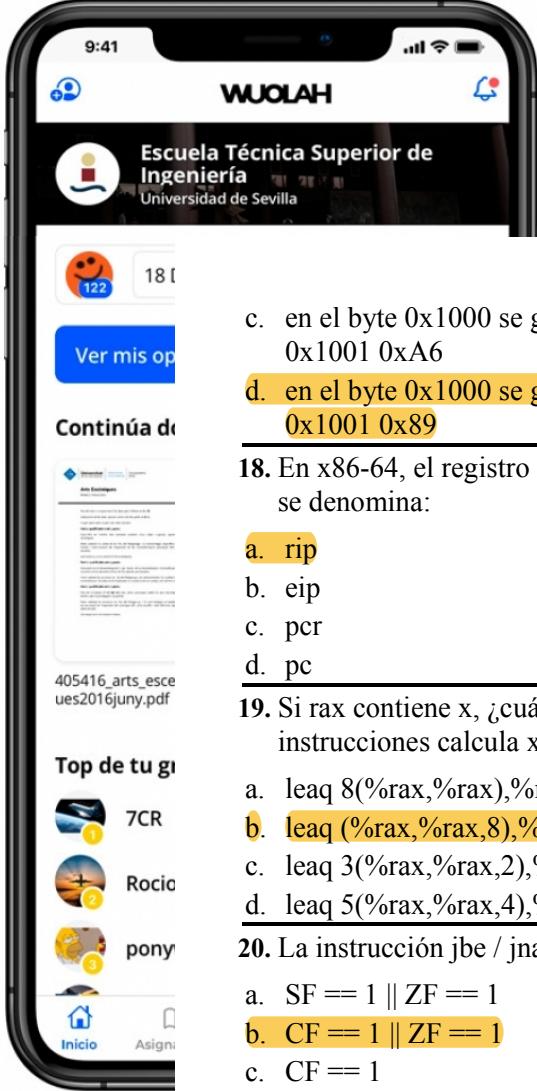
```
leaq (%rdi,%rdi,4), %rax
addq %rax, %rsi
movl var(%rsi,4), %eax
ret
```

Se puede adivinar que:

- a. var es un array multi-nivel (punteros a enteros) de cuatro filas



- b. var es un array bidimensional de enteros, con cinco columnas
- c. var es un array multi-nivel pero no se pueden adivinar las dimensiones
- d. var es un array bidimensional de enteros, no se pueden adivinar dimensiones
-
8. El microcódigo de un procesador consiste en 640 microinstrucciones de 70 bits, de las cuales 280 son únicas. ¿Cuántos bits se podría llegar a ahorrar usando nanoprogramación?
- a. $640 \times 70 - 280 \times 70$
- b. **640 x 70 - 640 x 9 - 280 x 70**
- c. $640 \times 70 - (640 \times \log_2(70) + 280 \times 70)$
- d. $640 \times 280 - 640 \times 70 - 280 \times 70$
-
9. ¿A qué instrucción de lenguaje máquina podría corresponder la siguiente secuencia de microinstrucciones del camino de datos con un bus estudiado en clase?
- Enable R1, Load Y
Enable R2, Select Y, Add, Load Z
Enable Z, Load R2*
- a. load (R1,R2), R2
- b. **add R1, R2**
- c. store R1+R2, (R2)
- d. move Y(R1,R2), Z(R2)
-
10. Suponer que un procesador ideal que ejecuta cada instrucción en T segundos se segmenta en cuatro etapas ideales de duración T/4. Con ello se consigue que
- a. 4 instrucciones se ejecuten en 4T segundos
- b. una instrucción se ejecute en T/4 segundos
- c. cada 4T segundos se terminen de ejecutar 4 instrucciones
- d. **cada T/4 segundos se termine de ejecutar una instrucción**
-
11. Precaptar instrucciones antes de que sean necesarias y almacenarlas en una cola de instrucciones, es una técnica que se usa para...
- a. **evitar cierto tipo de riesgos estructurales**
- b. reducir riesgos por dependencias de datos
- c. corregir algunos riesgos de control
- d. calcular las predicciones de saltos
-
12. La técnica de Consulta de Estado (polling) puede usarse para... (señalar la opción INCORRECTA)
- a. identificar el origen de una interrupción
- b. consultar si el dispositivo está dispuesto para entregar o recibir datos
- c. **consultar el sentido del DMA en curso (desde memoria / hacia memoria)**
- d. establecer un mecanismo software de asignación de prioridades a los dispositivos
-
13. Técnicas que suelen contemplar los procesadores para gestionar el sistema de prioridades entre peticiones de interrupción (señalar la opción INCORRECTA)
- a. gestión de prioridades centralizada
- b. gestión de prioridades distribuida
- c. **gestión de prioridades simultánea**
- d. gestión de prioridades híbrida
-
14. Una SRAM de 256Kx4bit (1Mbit) puede venir organizada en 1024 filas, dedicando por tanto al decodificador de columnas...
- a. 6 bits
- b. 7 bits
- c. **8 bits**
- d. 10 bits
-
15. ¿Cuál de los siguientes grupos de señales no se usa en un chip de memoria SRAM?
- a. Selección de chip CS# y habilitación de escritura WE#
- b. **Selección de filas RAS# y de columnas CAS#**
- c. Direcciones A_{n-1}-A₀
- d. Datos D_{n-1}-D₀
-
16. ¿En qué pareja de registros están la dirección de memoria y el dato que se leerá o escribirá en memoria?
- a. pc y mar
- b. ir y pc
- c. **mar y mdr/mbr**
- d. mdr/mbr y pc
-
17. Si queremos almacenar la palabra de 16 bits 0x8965 en memoria según little-endian, quedará almacenada a partir de la posición 0x1000 como:
- a. en el byte 0x1000 se guarda 0xA6 y en el 0x1001 0x91
- b. en el byte 0x1000 se guarda 0x89 y en el 0x1001 0x65



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- c. en el byte 0x1000 se guarda 0x91 y en el 0x1001 0xA6
- d. en el byte 0x1000 se guarda 0x65 y en el 0x1001 0x89

18. En x86-64, el registro contador de programa se denomina:

- a. rip
- b. eip
- c. pcr
- d. pc

19. Si rax contiene x, ¿cuál de las siguientes instrucciones calcula x*9?

- a. leaq 8(%rax,%rax),%rdx
- b. leaq (%rax,%rax,8),%rdx
- c. leaq 3(%rax,%rax,2),%rdx
- d. leaq 5(%rax,%rax,4),%rdx

20. La instrucción jbe / jna provoca un salto si...

- a. SF == 1 || ZF == 1
- b. CF == 1 || ZF == 1
- c. CF == 1
- d. SF != OF

21. En x86-64, es responsabilidad del procedimiento llamado (calle) salvaguardar, entre otros, los registros:

- a. %rbx, %rsi, %rdi
- b. %rax, %rdx, %rcx
- c. %rax, %rbx, %rcx, %rdx
- d. %rbx, %rbp

22. En x86-64, una función con 10 parámetros de tipo long que devuelve el valor del 8º parámetro y no modifica el puntero de pila puede traducirse a ensamblador como:

- a. movq 8(%rsp), %rax
ret
- b. movq 16(%rsp), %rax
ret
- c. movq %r10, %rax
ret
- d. movq %r8, %rax
ret

23. ¿Cuál de las siguientes características es típica de la microprogramación horizontal?

- a. Muchos campos solapados
- b. Poca codificación
- c. Microinstrucciones cortas

- d. Poca capacidad para expresar paralelismo entre microoperaciones

24. En un camino de datos con un solo bus, para realizar la operación de copia de un registro r1 en un registro r2, es decir $r2 \leftarrow r1$, es necesario:

- a. Activar la carga del registro r1 y habilitar la salida triestado del registro r2
- b. Habilitar la salida triestado del registro r1 y activar la carga del registro r2
- c. Habilitar las salidas triestado de los registros r1 y r2 y activar la carga del registro r2
- d. Habilitar la salida triestado del registro r2 y activar la carga de los registros r1 y r2

25. Sobre la segmentación:

- a. La frecuencia de reloj viene impuesta por la etapa más corta.
- b. Existen limitaciones al rendimiento provocadas por las instrucciones de salto y por las dependencias de datos.
- c. Es una técnica para lanzar a ejecutar simultáneamente varias instrucciones con el fin de reducir el tiempo de ejecución.
- d. Un procesador superescalar no puede estar segmentado.

26. La ganancia de velocidad ideal en un cauce de K etapas de igual duración T ejecutando un programa de N instrucciones es:

- a. $S=(K \cdot N)/(K-N+1)$
- b. $S=(N \cdot K \cdot T)/((N-K+1) \cdot T)$
- c. $S=(K \cdot N)/(K+N-1)$
- d. $S=(N \cdot T)/((N+K-1) \cdot T)$

27. ¿Cuál de las siguientes tareas NO es responsabilidad de un circuito de interfaz o controlador de periféricos sencillo?

- a. Adaptar el formato de las señales.
- b. Ajustar la temporización entre el procesador y los dispositivos de E/S.
- c. Recibir señales de control desde el procesador.
- d. Ejecutar el programa de transferencia de información entre el procesador y los dispositivos de E/S.

28. ¿Con cuál de los siguientes dispositivos tendría sentido utilizar E/S programada sin consulta de estado?

- a. Salida a un display de 7 segmentos
- b. Entrada desde un disco duro
- c. Salida a una impresora
- d. Con ningún dispositivo tiene sentido
-
29. ¿A qué tipo de localidad de memoria hace referencia la siguiente afirmación: "si se referencia un elemento, los elementos cercanos a él serán referenciados pronto"?
- a. Localidad espacial
- b. Localidad secuencial
- c. Localidad temporal
- d. Localidad asociativa
-
30. ¿Cuál de las siguientes afirmaciones acerca de la memoria es FALSA?
- a. La memoria dinámica usa señales de control RAS# y CAS#.
- b. Las celdas de memoria dinámica están constituidas por un transistor y un condensador.
- c. Las celdas de memoria estática tienen que ser constantemente refrescadas.
- d. La memoria estática se emplea en las caches L1 y L2.
-

Nombre:
DNI:
Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros sin signo en doble precisión acumulando acarreos. Para aplicar esta técnica se sugiere utilizar la(s) instrucción(es)...

- a. ADC \$0, ...
 - b. CLTD, ADD y ADC
 - c. JNC y INC
 - d. CLTQ, CQTO y IDIV
-

2. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros sin signo en doble precisión extendiendo con ceros. Para aplicar esta técnica se sugiere utilizar la(s) instrucción(es)...

- a. ADC \$0, ...
 - b. CLTD, ADD y ADC
 - c. JNC y INC
 - d. CLTQ, CQTO y IDIV
-

3. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros en doble precisión extendiendo el signo. Para aplicar esta técnica se sugiere utilizar la(s) instrucción(es)...

- a. ADC \$0, ...
 - b. CLTD, ADD y ADC
 - c. JNC y INC
 - d. CLTQ, CQTO y IDIV
-

4. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount5:

```
int popcount5(int* array, size_t len) {
    size_t i,j,val;
    long x;
    int result=0;
    for (i=0; i<len; i++) {
        x = array[i];
        val = 0;
        for (j=0; j<8; j++) {
            val += x & 0x0101010101010101L;
            x >>= 1;
        }
        val += (val >> 32);
        val += (val >> 16);
        val += (val >> 8);
        result+= val & 0xFF;
    }
    return result;
}
```

Esta función no sigue la adaptación a 32 bits recomendada en clase como versión "oficial", y conserva muchos vestigios de la versión de 64 bits, como los tipos de **array**, **x**, **val**, la máscara **0x...01L** y el desplazamiento **>>32**.

Esta función popcount5 fallaría:

- a. con array={0x80000000, 0x00400000, 0x00000200, 0x00000001}
 - b. con array={0, -1, -2, -3}
 - c. con ambos ejemplos
 - d. con ninguno de los dos ejemplos
-

5. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount6:

```
int pc6(unsigned* array, size_t len) {
    size_t i;
    unsigned x;
```



```

int result=0;

const unsigned m1 = 0x55555553;
const unsigned m2 = 0x33333333;
const unsigned m4 = 0x0f0f0f0f;
const unsigned m8 = 0x00ff00ff;
const unsigned m16 = 0x0000ffff;

for (i=0; i<len; i++) {
    x = array[i];

    x = (x & m1) + ((x >> 1) & m1);
    x = (x & m2) + ((x >> 2) & m2);
    x = (x & m4) + ((x >> 4) & m4);
    x = (x & m8) + ((x >> 8) & m8);
    x = (x & m16) + ((x >> 16) & m16);

    result+= x;
}
return result;
}

```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en algún dígito de alguna máscara.

Esta función `popcount6` fallaría:

- a. con `array={0x80000000, 0x00400000, 0x00000200, 0x00000001}`
- b. con `array={0x10000000, 0x00200000, 0x00000400, 0x00000008}`**
- c. con ambos ejemplos
- d. con ninguno de los dos ejemplos

6. En una bomba como las estudiadas en prácticas, del tipo...

```

400746: lea    0xc(%rsp),%rsi
40074b: lea    0x1ab(%rip),%rdi
              # 4008fd <stdin_u+0x10d>
400752: mov    $0x0,%eax
400757: callq  400590 <scanf@plt>
40075c: cmpl   $0x400600,0xc(%rsp)
400764: je     40076b <main+0x8c>
400766: callq  400697 <boom>
40076b: callq  4006bb <defused>

```

...el código numérico (pin) es...

- a. el entero 0x400590
- b. el entero 0x400600**
- c. el entero almacenado a partir de `0xc(%rsp)`
- d. el entero alm. a partir de `0x1ab+0x400752`

7. En la práctica "blink" de Arduino, un estudiante muestra a los profesores lo que le han enseñado en algún otro lugar: conectando un led directamente entre las patillas *Digital pin 13 (LED_BUILTIN)* y GND (justo al lado) también parpadea. Cabe esperar la siguiente respuesta de los profesores:

- a. Exactamente esa solución viene en el guión, no hace falta aprenderla fuera de clase
- b. Nosotros conectamos los led en serie con una resistencia como dice el guión**
- c. Hay que revisar ese equipo en concreto, debe tener algún defecto, ese led no debe parpadear
- d. Es cierto, es un error de diseño del guión, es más sencillo conectarlo así

8. Suponga una memoria cache con las siguientes propiedades: Tamaño: 32 KB. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Tamaño de línea 64 B. Dado el siguiente fragmento de código:

```

int x[262144], y[262144], a=0;
for (i = 0; i < 262144; i++)
    a += x[i] * y[i];

```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?

Suponer que el tamaño de los tipos de datos es como en x86-64, y que el compilador optimiza el acceso a las variables `a/i` en un registro.

- a. 1/2 (un fallo por cada 2 accesos)
- b. 1/4 (un fallo por cada 4 accesos)
- c. 1/8 (un fallo por cada 8 accesos)
- d. 1/16 (un fallo por cada 16 accesos)**

9. En el programa `line.cc` de la práctica "cache", se accede al vector saltando...

- a. de byte en byte
- b. de 64 en 64 bytes
- c. de 1 KB en 1 KB
- d. de line en line bytes, donde line barre los valores desde 1 hasta 1K en un bucle for**

10. En la práctica de cache hemos hecho una gráfica con el código `line.cc` ¿Qué forma tiene la gráfica que se puede obtener?

- a. Forma de U (o V), con un tramo descendente y otro ascendente
- b. Forma de /, una gráfica siempre creciente
- c. Forma de media U seguida de \, es decir, un tramo descendente suave, un pequeño tramo horizontal, y un tramo descendente lineal**
- d. Una escalera con varios tramos horizontales

11. ¿Cuál de las siguientes instrucciones es incorrecta?

- a. `shr $1,%rdx`
- b. `shr %rdx`
- c. `shr %cl,%rdx`**

ENCENDER TU LLAMA CUESTA MUY POCO



d. shr %rcx,%rdx

12. En la práctica “media” se pide sumar una lista de 16 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista de 16 enteros causaría overflow con 32bits?

PISTA: Sumar un número 16 veces == multiplicarlo por 16 == desplazarlo 4 bits a la izquierda

- a. 0xffff ffff
- b. 0xfc00 0000
- c. 0xf7ff ffff
- d. 0xf000 0000

13. La función siguiente, que aparece en el esqueleto suma_09 presentado en la práctica “popcount”

```
int suma3(int* array, int len){  
    asm("mov $0, %eax\n"  
        "    mov $0, %%rdx\n"  
        "bucle:\n"  
        "    add (%%%rdi,%%%rdx,4), %%eax\n"  
        "    inc %%rdx\n"  
        "    cmp %%rdx,%%rsi\n"  
        "    jne bucle\n"  
        "        :\n"           // output  
        "        :\n"           // input  
        : "cc",\n          "rax", "rdx"\n        : "clobber"\n    );  
}
```

no incluye la instrucción **ret** dentro de la sentencia **asm** porque:

- a. porque la función no devuelve nada
- b. porque suma3 es un puntero a función
- c. porque la incluye automáticamente el compilador
- d. porque suma3 no se llamará con call sino con jmp

14. Respecto a la inicialización de los elementos del array en el esqueleto suma_09 de la práctica “popcount”:

```
#define SIZE (1<<16)  
for (i=0; i<SIZE; i++)  
    lista[i]=i;
```

- a. Cualquiera de las tres funciones suma del programa suma_09_Casm irán más rápido con esta inicialización que con números aleatorios.
- b. Las tres funciones suma del programa suma_09_Casm irán igual de rápido con esta inicialización que con números aleatorios.

- c. popcount1 (bucle interno for) irá más rápido con esta inicialización que con números aleatorios
- d. popcount2 (bucle interno while) irá más lento con esta inicialización que con números aleatorios.

15. Un estudiante entrega la solución de una bomba y en ella aparece el comando de gdb “**p(char*)\$rdi**”. Este comando:

- a. muestra únicamente el contenido de %dil
- b. es erróneo sintácticamente
- c. muestra el valor de la dirección de memoria contenida en %rdi y el string almacenado a partir de esa dirección de memoria
- d. muestra el contenido de %rdi interpretado como un string de 8 caracteres

16. En la realización de la práctica de la bomba digital compilada en 32 bits, una parte del código máquina es el siguiente:

```
0x080486e8: call 0x8048524 <strncmp>  
0x080486ed: test %eax,%eax  
0x080486ef: je 0x80486f6 <main+134>  
0x080486f1: call 0x8048604 <boom>
```

¿Cuál de los siguientes comandos de gdb cambiaría el salto condicional por un salto incondicional?

- a. set \$0x080486ef=0xeb
- b. set *(char*)0x080486ef=0xeb
- c. set *(char*)0x080486f6=jmp
- d. set %0x080486ef=0xeb

17. ¿Cuál de las siguientes patillas (pin) no se menciona (no se pide usar/no se debe usar) en la 1ª práctica del Theremín de luz de Arduino? (Proyecto p06 / sólo fotocélula y zumbador)

- a. Analog In A0
- b. AREF
- c. ledPin = 13
- d. GND

18. En la práctica del Theremín de luz de Arduino se pide combinar el código del Proyecto p06 (sólo fotocélula y zumbador) con el de la Lección 26 (fotocélula y leds). Para conseguir que se enciendan y apaguen correctamente todos los led, lo mejor sería:

- a. Reutilizar la variable pitch calculada mediante map() como la nueva numLEDSLit

- b. Fusionar las variables `sensorValue` y `reading` de forma que sólo se llame una vez a `analogRead()`
- c. Calcular `numLEDSLit` con otra función `map()` de 0 a 8, en lugar de barrer de 50 a 4000
- d. Copiar uno y otro código en las secciones correspondientes, no se puede fusionar nada

19. El código de `size.cc` accede al vector saltando de 64 en 64. ¿Por qué?

- a. Porque cada elemento del vector ocupa 64 bytes.
- b. Para recorrer el vector más rápidamente.
- c. Porque el tamaño de cache L1 de todos los procesadores actuales es de 64 KB.
- d. Para anular los aciertos por localidad espacial, es decir, para que sólo pueda haber aciertos por localidad temporal.

20. Un servidor tiene dos procesadores Intel Xeon E5-2620 v3@ 2.40GHz (2,4 Ghz, 6 núcleos, 12 hebras, 15MiB de caché L3, reloj DDR4 a 1866 MHz). Suponga que un proceso, que se ejecuta en un único núcleo, tiene que ordenar un vector de datos de usuarios accediendo repetidamente a sus elementos. Cada elemento es una estructura de datos y tiene un tamaño de 4 KB. Según <http://www.cpu-world.com/> los tamaños de cache son:

Cache: L1 data L1 instr. L2 L3
Size: 6 x 32KB 6 x 32KB 6 x 256KB 15MB

Si representamos en una gráfica las prestaciones en función del número de usuarios a ordenar, ¿para qué límites teóricos en el número de usuarios se observarán saltos en las prestaciones debidos a accesos a la jerarquía de memoria?

- a. 4 / 32 / 512 usuarios
- b. 8 / 64 / 3840 usuarios
- c. 48 / 384 / 23040 usuarios
- d. 32 / 256 / 8192 usuarios

Nombre:

DNI:

Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. En el direccionamiento inmediato el operando reside en:
 - a. en un registro del procesador
 - b. en la instrucción tras el código de operación**
 - c. en memoria, en la dirección indicada
 - d. en la pila

2. Una instrucción máquina del tipo "Add M,R" podría formar parte del repertorio de
 - a. una máquina pila
 - b. una máquina de acumulador
 - c. una máquina con arquitectura R/R
 - d. una máquina con arquitectura M/M**

3. La traducción a ASM de una función C con prototipo **void fun(<tipo> arg1, <tipo> arg2);** por parte de gcc empieza con el siguiente texto:


```
fun:
    movl (%rdi), %eax
    movl (%rsi), %edx
    ...
  
```

¿Cuál es el posible <tipo> mencionado en el prototipo?

 - a. long *
 - b. unsigned int ***
 - c. short *
 - d. unsigned char *

4. Se puede describir paso a paso la ejecución de la instrucción **add (%rbx, %rdx,** **4), %eax;** de la siguiente manera (marcar la opción FALSA):

(marcar la opción FALSA):

- a. primero: se calcula la dirección efectiva $EA = RBX + RDX * 4$
- b. segundo: se leen ocho bytes a partir de dicha dirección de memoria**
- c. tercero: se yuxtaponen los bytes leídos de forma que el de dirección más baja ("el primero leído") sea el menos significativo y el de dirección más alta ("el último leído") sea el más significativo
- d. cuarto: se suma el valor obtenido con el registro EAX

5. ¿Qué valor contendrá el registro rdx tras ejecutar las dos instrucciones siguientes?

```
movq $-1, %rdx
movl $1, %edx
```

- a. 0xFFFF FFFF FFFF 0001
 - b. 0xFFFF FFFF 0000 0001
 - c. 0xFFFF 0000 0000 0001
 - d. 0x0000 0000 0000 0001**
-

6. En el fragmento de código


```
400544: e8 07 00 00 00 callq 400550 <f>
400549: 48 89 03          mov %rax, (%rbx)
```

la instrucción call suma al contador de programa la cantidad:

 - a. 0x00000007**
 - b. 0x00400549
 - c. 0x400544
 - d. 0x48

- 7. Una función C declarada como **int get_var_digit(size_t index, size_t digit)** genera como código ensamblador

```

movq var(,%rdi,8), %rax
movl (%rax,%rsi,4), %eax
ret

```

Se puede adivinar que:

- a. var es un array multi-nivel (punteros a enteros) de cuatro filas
 - b. var es un array multi-nivel pero no se pueden adivinar las dimensiones**
 - c. var es un array bidimensional de enteros, con ocho columnas
 - d. var es un array bidimensional de enteros, con cinco columnas
-
8. En una unidad de control microprogramada se tiene un campo de 14 señales de control de las cuales sólo se activaría una o ninguna en un ciclo de reloj, nunca dos o más en el mismo ciclo de reloj. Sería entonces posible...
 - a. codificarlas con 4 bits, y sobraría un código que quedaría sin uso**
 - b. codificarlas con 5 bits, y sobrarían dos códigos que quedarían sin uso
 - c. solaparlas en un solo campo de 5 bits, ahorrando por tanto 9 bits
 - d. solaparlas en un solo campo de 7 bits, ahorrando por tanto 6 bits
-
9. En el contexto de microprogramación, el control residual...
 - a. intenta disminuir la cantidad de "bits residuales", usando las técnicas de codificación y/o solapamiento de campos, como opuestas a la microprogramación directa o "inmediata".
 - b. se refiere a que cuanto más codificación y/o solapamiento se use, menos capacidad para expresar paralelismo se tiene, siendo ese menor control un "residuo" o consecuencia no deseada de dichas técnicas no "inmediatas"
 - c. clasifica las microinstrucciones del microcódigo según formen parte de micropogramas ("microinstr. inmediatas") o no ("microinstr. residuales")
 - d. consiste en almacenar señales de control en un "registro de control residual" para usarlas en ciclos posteriores, a diferencia del "control inmediato", en donde los bits se utilizan inmediatamente**

10. Suponer que un procesador ideal que ejecuta cada instrucción en T segundos se segmenta en cuatro etapas ideales de duración T/4. ¿Cuál razonamiento es correcto?

- a. Se espera una reducción de prestaciones porque además de ejecutar las instrucciones hay que segmentarlas (coste de la segmentación)**
 - b. Se espera una reducción de prestaciones porque la duración del ciclo de reloj vendrá impuesta por la etapa más lenta
 - c. Se espera un aumento de prestaciones debido al efecto de los riesgos (hazards) sobre el avance de las instrucciones en el cauce
 - d. Se espera un aumento de prestaciones debido a que las cuatro etapas solapan su funcionamiento, con una aceleración ideal de 4x**
-

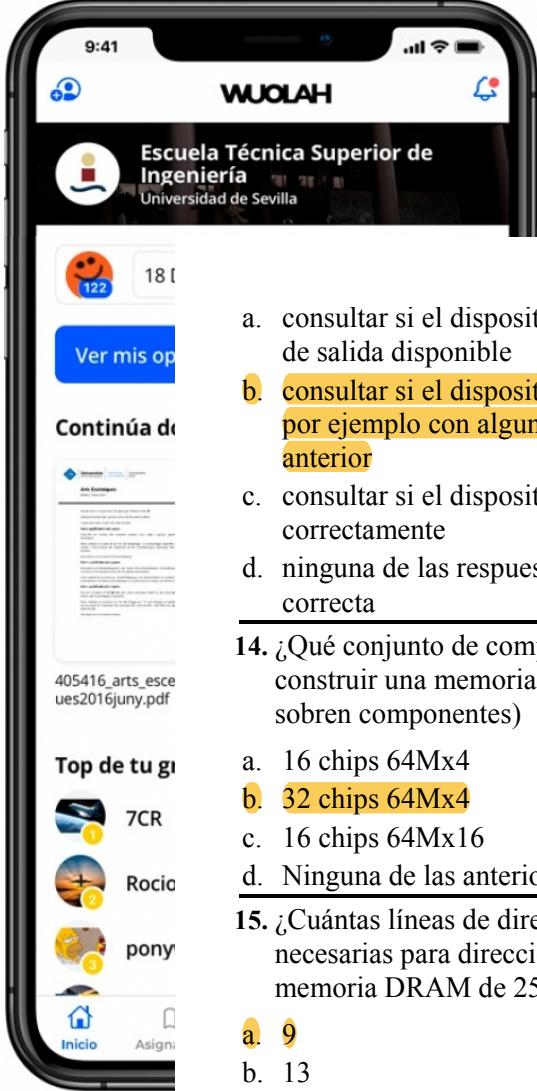
11. Un salto condicional del tipo "delayed branch", o salto retardado, ejecuta la(s) instrucción(es) siguiente(s)...

- a. sólo si el salto se produce (las ignora si NO se produce), de manera que instrucción(es) en el destino del salto podrían adelantarse tras la propia instrucción de salto
 - b. sólo si el salto NO se produce (las ignora si se produce), de manera que instrucción(es) en el destino del salto podrían adelantarse tras la propia instrucción de salto
 - c. siempre, de manera que instrucción(es) anterior(es) al salto podrían colocarse tras la propia instrucción de salto**
 - d. nunca, de manera que instrucción(es) anterior(es) al salto no podrían colocarse tras la propia instrucción de salto
-

12. Un computador con 20 líneas de dirección y memoria de bytes tiene 640KB de RAM, 128KB de ROM, y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 32 direcciones?

- a. 2^{10}
 - b. 2^{11}
 - c. 2^{12}
 - d. 2^{13}**
-

13. La consulta de estado que se puede llevar a cabo en una operación de salida mediante E/S programada sirve para...



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- a. consultar si el dispositivo tiene algún dato de salida disponible
- b. **consultar si el dispositivo está aún ocupado, por ejemplo con alguna operación de salida anterior**
- c. consultar si el dispositivo funciona correctamente
- d. ninguna de las respuestas anteriores es correcta

14. ¿Qué conjunto de componentes permite construir una memoria 256Mx32? (sin que sobren componentes)

- a. 16 chips 64Mx4
- b. **32 chips 64Mx4**
- c. 16 chips 64Mx16
- d. Ninguna de las anteriores

15. ¿Cuántas líneas de dirección (patillas) son necesarias para direccionar un chip de memoria DRAM de 256K x 4?

- a. **9**
- b. 13
- c. 18
- d. 22

16. ¿Qué arquitectura es típica en procesadores RISC?

- a. registro-registro
- b. registro-memoria
- c. memoria-registro
- d. memoria-memoria

17. ¿Cuál de las siguientes características es posterior a la segunda generación de computadores?

- a. Memoria de núcleos de ferrita
- b. Compilador
- c. **Memoria cache**
- d. Lenguaje ensamblador

18. Si el registro rax contiene **x**, la sentencia en **C x &= 0x1;** se traducirá a ensamblador como:

- a. **andq \$1, %rax**
- b. orq \$0x1, %rax
- c. shrq %rax
- d. sarq %rax

19. Para crear espacio en la pila para variables locales sin inicializar suele realizarse la siguiente operación:

- a. Restar una cantidad positiva a rbp.
- b. Sumar una cantidad positiva a rbp.
- c. **Restar una cantidad positiva a rsp.**
- d. Sumar una cantidad positiva a rsp.

20. Si la estructura struct a ocupa un espacio de 26 bytes en memoria, ¿cuántos bytes ocupa la siguiente estructura struct b cuando se compila en 64 bits?

```
struct b {  
    struct a a1;  
    int i;  
    struct a a2;  
};
```

- a. 24
- b. 58
- c. **60**
- d. 64

21. En la secuencia de programa siguiente:

```
628: e8 cd ff ff ff  callq  5fa <suma>  
62d: 48 83 c4 20    add    $0x20,%rsp
```

¿cuál es el valor que introduce en la pila la instrucción callq?

- a. 0xffffffffcd
- b. 0x5fa
- c. 0x628
- d. **0x62d**

22. Un archivo .o que contiene código objeto reubicable:

- a. **Contiene instrucciones máquina binarias.**
- b. Contiene instrucciones máquina y directivas en ensamblador.
- c. Puede ejecutarse directamente.
- d. Contiene las direcciones definitivas de las variables globales.

23. ¿Cuál de las siguientes sentencias sobre la unidad de control es FALSA?

- a. Cuanto más horizontal es la microprogramación, más largas son las microinstrucciones
- b. Debido al pequeño número de operaciones simples, la sección de control de un procesador RISC puede ser cableada en lugar de microprogramada
- c. **El programador de lenguaje ensamblador necesita conocer la microarquitectura del ordenador**

- d. Es posible realizar el diseño físico de una unidad de control cableada de manera semiautomática
-
- 24.** En una unidad de control microprogramada con formato de microinstrucciones vertical, un subcampo que deba especificar 16 señales de control, codificadas de tal forma que pueda activarse sólo una o ninguna señal de control, habrá de tener una anchura mínima de:
- 4 bits
 - 5 bits**
 - 16 bits
 - 17 bits
-
- 25.** La técnica de "adelanto de registros" (register forwarding) en un cauce segmentado se usa para limitar el impacto de los riesgos...
- estructurales
 - organizativos
 - de control
 - de datos**
-
- 26.** Un sistema no segmentado tarda 10 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 4 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se acerca a:
- 10
 - 4
 - 40
 - 2,5**
-
- 27.** ¿Cuál de las siguientes afirmaciones es cierta?
- La E/S en memoria emplea la patilla IO/M#.
 - En E/S independiente, las instrucciones de acceso a memoria suelen ser más largas que las de E/S.**
 - La E/S en memoria facilita la protección.
 - En la E/S en memoria las instrucciones de E/S son fácilmente reconocibles.
-
- 28.** La instrucción máquina di (Disable Interrupts), conocida como cli (Clear Interrupt Flag) en x86:
- Desactiva todas las interrupciones enmascarables.**
-

b. Desactiva las interrupciones de inferior o igual prioridad a una dada.

c. Desactiva determinados niveles de interrupción de forma selectiva.

d. Desactiva las interrupciones software.

29. ¿Cuál de las siguientes afirmaciones sobre la jerarquía de memoria es cierta?

- Para aumentar la eficiencia se transfieren bloques completos.**
- Toda la información que el procesador necesita está en el nivel 1.
- Si una palabra no se encuentra en el tercer nivel entonces se busca en el segundo nivel.
- A medida que nos alejamos del procesador, el tamaño de memoria disminuye.

30. ¿Cuántas líneas de dirección (patillas) son necesarias para direccionar un chip de memoria DRAM de 4096 x 4?

- 6**
- 10
- 11
- 12

Nombre:

DNI:

Grupo:

Test de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.2p si es correcta, 0 si está en blanco o claramente tachada, -0.06p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros SIN signo produciendo un resultado .quad (doble precisión). El programa esqueleto ofrecido (suma.s, que suma la conocida lista de 9 elementos en simple precisión) no es válido. Se podría comprobar imprimiendo el resultado SIN signo y usando como contraejemplo las siguientes listas:

- a. 0x4000 0000 y 0x8000 0000 (el usuario piensa que sus datos NO tienen signo)
- b. 0x8000 0000 y 0xC000 0000
- c. ninguna de las dos
- d. ambas

2. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros CON signo produciendo un resultado .quad (doble precisión). El programa esqueleto ofrecido (suma.s) no es válido. Se podría comprobar imprimiendo el resultado SIN signo, y usando como contraejemplo las siguientes listas:

- a. 1 y -1 (el usuario piensa que sus datos tienen signo)
- b. 0x4000 0000 y 0x4000 0000
- c. ninguna de las dos
- d. ambas

3. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros CON signo produciendo un resultado .quad (doble precisión). El programa esqueleto ofrecido (suma.s) no es válido. Se podría comprobar imprimiendo el resultado CON signo, y

usando como contraejemplo las siguientes listas:

- a. -1 y -1 (el usuario piensa que sus datos tienen signo)
 - b. 0x4000 0000 y 0x4000 0000
 - c. ninguna de las dos
 - d. ambas
-
4. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. El famoso estudiante entrega la siguiente versión de `popcount5`:

```

int pcnt5(unsigned* array, size_t len) {
    size_t i,j;
    int x;
    int val, result=0;

    for (i=0; i<len; i++) {
        x = array[i];
        val = 0;
        for (j=0; j<=8; j++) {
            val += x & 0x01010101;
            x >>= 1;
        }
        val += (val >> 16);
        val += (val >> 8);
        result+= val & 0xFF;
    }
    return result;
}

```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en las condiciones del for interno y el tipo de x.

Esta función `popcount5` fallaría:

- a. con `array={1, 16, 256, 4096}`
- b. con `array={0x80000000, 0x00400000, 0x00000200, 0x00000001}`

- c. con ambos ejemplos
- d. con ninguno de los dos ejemplos

5. El estudiante vuelve a equivocarse de nuevo y entrega la siguiente versión de `popcount6`:

```
int pcnt6(unsigned* array, size_t len) {
    size_t i;
    unsigned x;
    int result=0;

    const unsigned m1 = 0x55555553;
    const unsigned m2 = 0x33333333;
    const unsigned m4 = 0x0f0f0f0f;
    const unsigned m8 = 0x00ff00ff;
    const unsigned m16 = 0x0000ffff;

    for (i=0; i<len; i++) {
        x = array[i];

        x = (x & m1) + ((x >> 1) & m1);
        x = (x & m2) + ((x >> 2) & m2);
        x = (x & m4) + ((x >> 4) & m4);
        x = (x & m8) + ((x >> 8) & m8);
        x = (x & m16) + ((x >>16) & m16);

        result+= x;
    }
    return result;
}
```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en algún dígito de alguna máscara.

Esta función `popcount6` fallaría:

- a. con `array={0xA0000000, 0x00B00000, 0x00000C00, 0x0000000D}`
- b. con `array={0x10000000, 0x00200000, 0x00000400, 0x00000008}`**
- c. con ambos ejemplos
- d. con ninguno de los dos ejemplos

6. Un fragmento del desensamblado de una "bomba" similar a las estudiadas en prácticas es:

```
400634: lea    0x4(%rsp),%rsi
400639: lea    0x1a4(%rip),%rdi
          # 4007e4 <_stdin+0xe4>
400640: mov    $0x0,%eax
400645: callq  4004c0 <scanf>
40064a: mov    0x4(%rsp),%eax
40064e: shr    %eax
400650: jc     40066a <L6>
400652: shr    %eax
400654: jnc   40066a <L6>
400656: shr    %eax
400658: jc     40066a <L6>
40065a: shr    %eax
40065c: jnc   40066a <L6>
40065e: shr    %eax
400660: jc     40066a <L6>
```

```
400662: shr    %eax
400664: jnc   40066a <L6>
400666: test   %eax,%eax
400668: jz    40066f <L7>
40066a: callq  4005c7 <boom>
40066f: callq  4005eb <defused>
```

El código numérico correcto (pin) es...

- a. 42**
 - b. el entero 0x40066A
 - c. el entero almacenado a partir de 0x4(%rsp)
 - d. ninguno de los anteriores
7. Sobre el resultado devuelto por la función de Arduino `map(sensorValue, sensorLow, sensorHigh, 50, 4000)`; usada en el programa del Theremín de luz:
- a. Si `sensorValue` vale 50, devuelve `sensorLow`
 - b. Si `sensorValue` vale 50, devuelve `sensorHigh`
 - c. Si `sensorValue` vale 2025, devuelve la mitad entre `sensorLow` y `sensorHigh`
 - d. Si `sensorValue` vale la mitad entre `sensorLow` y `sensorHigh`, devuelve 2025**

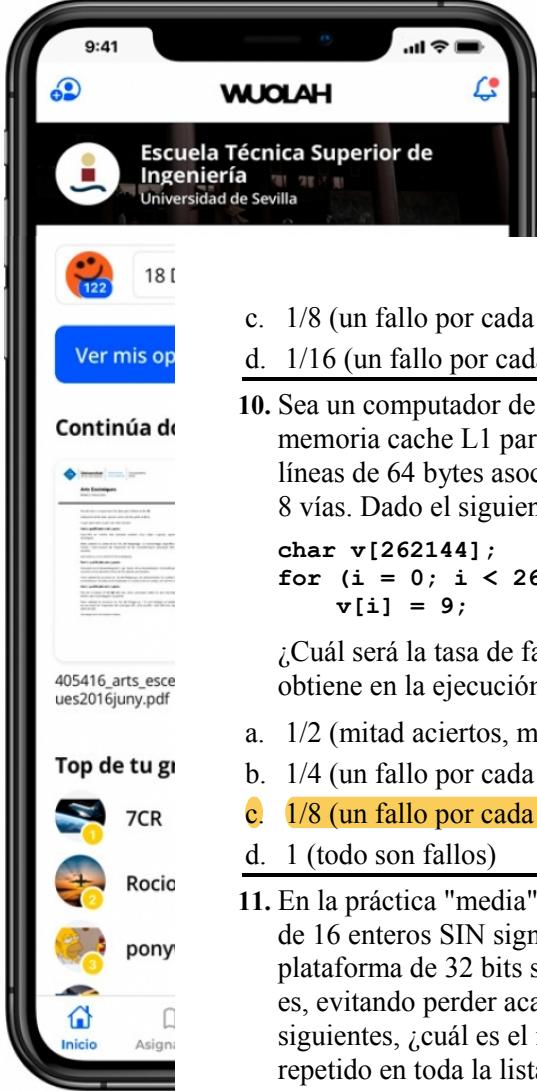
8. Suponga una memoria cache con las siguientes propiedades: Tamaño: 512 bytes. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Suponga que para la siguiente secuencia de direcciones enviadas a la cache: 0, 10, 16, 20, 30, 32, 40, 50, 60, 64, la tasa de acierto es 0,50. ¿Cuál es el tamaño de línea de la cache?
- a. 4 bytes
 - b. 8 bytes
 - c. 16 bytes**
 - d. Ninguno de los anteriores

9. Suponga una memoria cache con las siguientes propiedades: Tamaño: 32 KB. Política de reemplazo: LRU. Estado inicial: vacía (todas las líneas inválidas). Tamaño de línea 64 B. Dado el siguiente fragmento de código:

```
int v[262144], a=0;
for (i = 0; i < 262144; i += 8)
    a += v[i] * v[i+4];
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior? Suponer que el tamaño de los tipos de datos es como en x86-64, y que el compilador optimiza el acceso a las variables `a/i` en un registro.

- a. 1/2 (un fallo por cada 2 accesos)
- b. 1/4 (un fallo por cada 4 accesos)**



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

- c. 1/8 (un fallo por cada 8 accesos)
- d. 1/16 (un fallo por cada 16 accesos)

10. Sea un computador de 64 bits con una memoria cache L1 para datos de 32 KB y líneas de 64 bytes asociativa por conjuntos de 8 vías. Dado el siguiente fragmento de código:

```
char v[262144];
for (i = 0; i < 262144; i += 8)
    v[i] = 9;
```

¿Cuál será la tasa de fallos aproximada que se obtiene en la ejecución del bucle anterior?

- a. 1/2 (mitad aciertos, mitad fallos)
- b. 1/4 (un fallo por cada 4 accesos)
- c. 1/8 (un fallo por cada 8 accesos)**
- d. 1 (todo son fallos)

11. En la práctica "media" se pide sumar una lista de 16 enteros SIN signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando perder acarreos. De entre los siguientes, ¿cuál es el mínimo valor entero que repetido en toda la lista causaría acarreo con 32 bits (SIN signo)? Se usa notación decimal y espacios como separadores de millares / millones / etc.

- a. 10 000 000
- b. 100 000 000
- c. 1 000 000 000**
- d. 10 000 000 000

12. En la práctica "media" el mismo estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

```
bucle:
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %edx, %ebp
    jnc nocarry
    inc %edx
nocarry:
    inc %esi
    cmp %esi,%ecx
    jne bucle
```

Estando bien programado todo lo demás, este código...

- a. produce siempre el resultado correcto**
- b. fallaría con lista: .int 0,1,2,3
- c. fallaría con lista: .int -1,-2,-4,-8

- d. no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

13. Suponga la expresión: `unsigned int val = x & 0x01010101`; donde x es un unsigned int. ¿Cuál de los siguientes valores de x da como resultado val = 0x01010101?

- a. 0x10101010
- b. 0xaaaaaaaaaa
- c. 0xeeeeeeeeee
- d. 0x13579bdf**

14. En `popcount8`, la variable: `int SSE_LUTb[] = {0x02010100, 0x03020201, 0x03020201, 0x04030302};` se usa:

- a. como máscara para hacer un and con un registro xmm.
- b. como máscara para hacer un xor con un registro xmm.
- c. como tabla de consulta para reemplazar secuencias de 4 bits por sus correspondientes `popcount` de 4 bits.**
- d. como tabla de direcciones de registros xmm (por ej. 02 indica xmm2, 01 indica xmm1,...)

15. En una bomba como las estudiadas en prácticas, del tipo...

```
0x080486e8: call 0x8048524 <strncmp>
0x080486ed: test %eax,%eax
0x080486ef: je 0x80486f6 <main+134>
0x080486f1: call 0x8048604 <boom>
0x080486f6: ...
```

la contraseña correcta es...

- a. el string almacenado a partir de 0x8048524
- b. el string almacenado a partir de 0x80486f6
- c. el string almacenado a partir de 0x8048604
- d. ninguna de las otras opciones**

16. En una bomba que siga la misma estructura que la estudiada en el ejemplo dado en el guión de prácticas, y compilada en 32 bits:

```
0x0804873f: call 0x8048504 <scanf>
0x08048744: mov 0x24(%esp),%edx
0x08048748: mov 0x804a044,%eax
0x0804874d: cmp %eax,%edx
0x0804874f: je 0x8048756 <main+230>
0x08048751: call 0x8048604 <boom>
0x08048756: ...
```

...el código numérico correcto (pin) es...

- a. el entero 0x804a044

WUOLAH

- b. el entero cuya dirección está almacenada en la posición de memoria 0x804a044
- c. el entero almacenado a partir de la posición de memoria 0x24(%esp)
- d. el entero almacenado a partir de la posición de memoria 0x804a044
-
17. El circuito que hay que montar en la práctica del zumbador pasivo con Arduino es un:
- a. Buffer triestado
- b. Divisor de tensión
- c. Seguidor de voltaje
- d. Comparador
-
18. Al colocar un led en la placa de prototipado de Arduino, ¿cómo se sabe cuáles son el ánodo y el cátodo?
- a. Mirando las leyendas A+ o K- impresas en las patillas del led
- b. No es necesario saberlo, la polaridad es indiferente a la hora de colocar un led en un circuito
- c. Mirando los códigos de color dibujados sobre el encapsulado
- d. Mirando la longitud de las patillas o qué lado del encapsulado es plano
-
19. ¿Cuál de las siguientes afirmaciones sobre size.cc es cierta?
- a. La diferencia de velocidades entre L2 y L3 es mayor que la diferencia de velocidades entre L1 y L2.
- b. Si continuáramos multiplicando por 2 el tamaño del vector en el eje X obteniendo más puntos de la gráfica, esta continuaría horizontal para cualquier valor más allá de 64 MB.
- c. La gráfica tiene escalones hacia arriba porque en cada punto del eje X accedemos al mismo número de elementos del vector y el número de aciertos por localidad temporal disminuye bruscamente en ciertos puntos al aumentar el tamaño del vector.
- d. La gráfica tiene tramos horizontales porque el hecho de realizar la mitad de accesos al vector en cada punto de un tramo horizontal respecto al anterior punto de ese mismo tramo horizontal es compensado por el número de fallos creciente en ese mismo tramo horizontal.
-

20. ¿Qué forma tiene la gráfica que se debe obtener con el código size.cc?

- a. Forma de U, con un tramo descendente y otro ascendente.
- b. Forma de ∩, con un tramo ascendente y otro descendente.
- c. Una escalera decreciente con varios tramos horizontales.
- d. Una escalera creciente con varios tramos horizontales.
-

Nombre:
DNI:
Grupo:

Test de Teoría (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 0.1p si es correcta, 0p si está en blanco o claramente tachada, -0.03p si es errónea.

Anotar las respuestas (a, b, c ó d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. Un sistema con direcciones de 8bits utiliza una puerta NAND conectada a las líneas A7...A5 para atacar la entrada CS# (activa baja) de un módulo de memoria. En el mapa de memoria las siguientes posiciones corresponderán a dicho módulo

- a. 0x00 a 0x0f y 0x80 a 0x8f
- b. 0x00 a 0x1f
- c. 0x70 a 0x7f y 0xf0 a 0xff
- d. 0xe0 a 0xff**

2. ¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción x86-64 **add array(%rbx,4), %edx?**

- a. Direccionamiento a registro**
- b. Direccionamiento inmediato
- c. Direccionamiento indexado
- d. Direccionamiento relativo a registro base

3. ¿Qué valor contendrá el registro RDX tras ejecutar las dos instrucciones siguientes?

```
movq $-1, %rdx
shr $16, %edx
```

- a. 0xFFFF FFFF FFFF 0000
- b. 0xFFFF FFFF FFFF FFFF
- c. 0x0000 0000 0000 FFFF**
- d. 0xFFFF FFFF 0000 FFFF

4. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):

- a. movb %sil, (%rax)
- b. pushq \$0xFF
- c. movsbw (%rax), %dx

d. movzlx %edx, %rax

5. Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/x86-64 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como:
- a. var (el nombre de la variable representa su posición de memoria)
 - b. var(%rip)
 - c. k(%rsp), siendo k un número constante relativamente pequeño**
 - d. k(%rbp), siendo k un número constante positivo relativamente pequeño
6. Sabiendo que la instrucción de llamada mostrada abajo codifica la dirección de la subrutina con direccionamiento relativo a contador de programa (de 32 bits con signo), indicar el valor de los bytes tachados en el siguiente desensamblado.

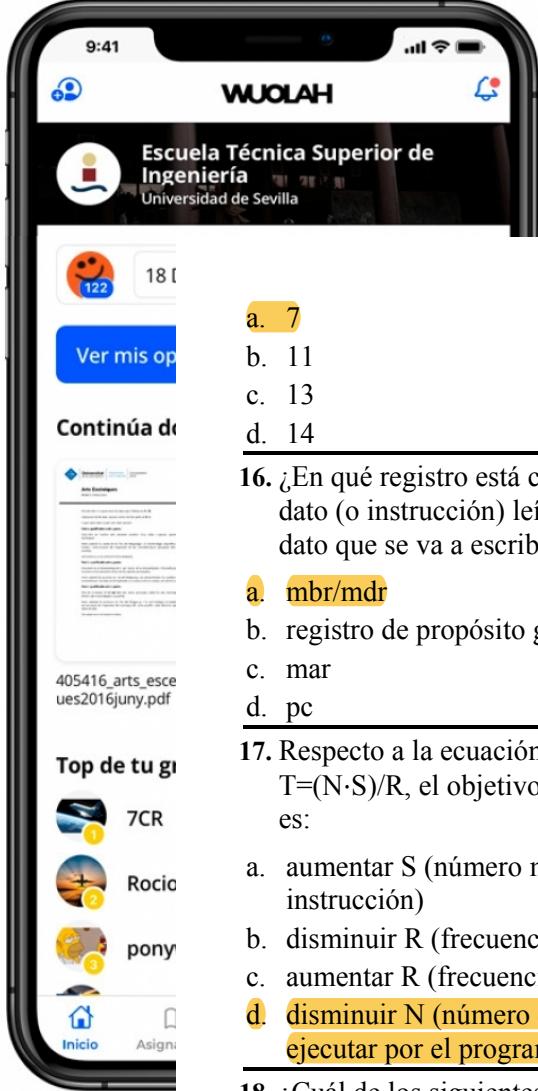
```
400544: e8 xx xx xx xx callq 400550 <ff>
400549: 48 89 03          mov %rax, (%rbx)
```

- a. 00 40 05 50
- b. 00 00 00 01
- c. 50 05 40 00
- d. 07 00 00 00**

7. Una función C declarada como **int *get_ap(struct rec *r, size_t idx)** genera como código ensamblador **leaq (%ordi,%rsi,4); ret**. Se puede adivinar que:
- a. la estructura contiene un array de 4 elementos



- b. la estructura comienza con un array de enteros (o tipo similar)
- c. la estructura contiene 5 campos
- d. r es un array de 4 estructuras
-
8. En una unidad de control microprogramada se tiene un campo de 14 señales de control de las cuales se activarían dos o menos en un ciclo de reloj, nunca tres o más en el mismo ciclo de reloj, y cuando se activan dos una es del grupo 1-7 y otra es del grupo 8-14, nunca las dos del mismo grupo. Sería entonces posible...
- a. codificarlas con 4 bits, y sobraría un código que quedaría sin uso
- b. codificarlas con 6 bits (2 códigos de 3 bits), y no sobraría ningún código sin uso**
- c. solaparlas en un solo campo de 5 bits, ahorrando por tanto 9 bits
- d. solaparlas en un solo campo de 7 bits, ahorrando por tanto 6 bits
-
9. En una unidad de control microprogramada se tienen dos campos de 5 señales de control mutuamente exclusivos, de manera que nunca se activarían señales de ambos campos en el mismo ciclo de reloj. Con esas 10 señales sería entonces posible...
- a. codificarlas con 4 bits, y sobraría un código que quedaría sin uso
- b. codificarlas con 5 bits, y no sobraría ningún código sin uso
- c. solaparlas en un solo campo de 5 bits, ahorrando por tanto 4 bits**
- d. solaparlas en un solo campo de 6 bits, ahorrando por tanto 5 bits
-
10. Suponer un procesador tipo MIPS segmentado en etapas IF, ID, EX, MEM, WB, como el presentado de ejemplo en clase. ¿Cuál afirmación es correcta?
- a. La aceleración del procesador no segmentado es 5x
- b. El periodo de reloj se escoge de acuerdo a la etapa más rápida del cauce
- c. Conviene tener memorias separadas de código y datos, para evitar conflictos entre las etapas IF y WB
- d. Conviene tener cache que permita acceso en un único ciclo de reloj**
-
11. Un salto condicional del tipo "annulling branch", o salto anulante, ejecuta la(s) instrucción(es) siguiente(s)...
- a. sólo si el salto se produce (las ignora si NO se produce), de manera que instrucción(es) en el destino del salto podrían adelantarse tras la propia instrucción de salto
- b. sólo si el salto NO se produce (las ignora si se produce), de manera que instrucción(es) en el destino del salto podrían adelantarse tras la propia instrucción de salto
- c. siempre, de manera que instrucción(es) anterior(es) al salto podrían colocarse tras la propia instrucción de salto
- d. nunca, de manera que instrucción(es) anterior(es) al salto no podrían colocarse tras la propia instrucción de salto
-
12. ¿Cuál de las siguientes técnicas no se puede usar para determinar la causa de una interrupción?
- a. Múltiples líneas de interrupción INT1#, INT2#...
- b. Línea de reconocimiento INTA#**
- c. Consulta de estado, o polling
- d. Interrupciones vectorizadas
-
13. ¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?
- a. Registro de dirección
- b. Registro de prioridades**
- c. Registro de órdenes (o de control)
- d. Registro contador
-
14. El primer nivel de una jerarquía de memoria tiene una tasa de aciertos del 80% y las peticiones de memoria tardan 15 ns en completarse si dicha posición se encuentra en ese nivel y 100 ns si no es así. ¿Cuál es el tiempo medio de acceso de la jerarquía?
- a. 23 ns
- b. 32 ns**
- c. 57.5 ns
- d. 92 ns
-
15. Un sistema tiene una cache asociativa por conjuntos de 8-vías con 64 conjuntos y líneas de 64B. ¿A qué conjunto le corresponde el byte con dirección 0xcafec1c0?



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.

Available on the
App Store

GET IT ON
Google Play

a. 7

- b. 11
- c. 13
- d. 14

16. ¿En qué registro está contenido el último dato (o instrucción) leído de memoria, o el dato que se va a escribir en memoria?

- a. mbr/mdr
- b. registro de propósito general
- c. mar
- d. pc

17. Respecto a la ecuación de rendimiento $T=(N \cdot S)/R$, el objetivo de un diseño CISC es:

- a. aumentar S (número medio de ciclos por instrucción)
- b. disminuir R (frecuencia de reloj)
- c. aumentar R (frecuencia de reloj)
- d. **disminuir N (número de instrucciones a ejecutar por el programa)**

18. ¿Cuál de los siguientes microprocesadores NO es de 64 bits?

- a. Core i3
- b. Core i7
- c. **AVR ATMega**
- d. Itanium

19. La instrucción movq %rsp,%rbp

- a. Intercambia los contenidos de los registros rsp y rbp.
- b. Introduce en la pila el contenido del registro rbp.
- c. **Copia el contenido del registro rsp en el registro rbp.**
- d. Mueve el contenido del registro rsp al registro rbp, poniendo a 0 el registro rsp.

20. Si desplazamos rbx mediante salq \$4, %rbx:

- a. Lo multiplicamos por 4
- b. Lo dividimos por 4
- c. **Lo multiplicamos por 16**
- d. Lo dividimos por 16

21. Si el registro rax contiene un long (64 bits CON signo) x, la secuencia de instrucciones siguiente:

```
cmpq $10, %rax
ja dest
```

saltará a la etiqueta dest si:

- a. $x > 9$
- b. $x < 10$
- c. **$x \leq -1 \text{ || } x \geq 11$**
- d. $x \geq 0 \text{ && } x \leq 10$

22. Se ha declarado en un programa C la variable **int val[5]={1,5,2,1,3};** ¿Cuál de las siguientes afirmaciones es cierta?

- a. $\text{val}[1] == 1$
- b. **&val[3] == (char *)val + 12**
- c. $\text{sizeof}(\text{val}) == 5$
- d. $\&\text{val}[2]$ es de tipo int * y vale lo mismo que $\text{val} + 8$

23. La conexión de las salidas de tres registros hacia un bus común en el camino de datos puede realizarse usando...

- a. **dos multiplexores de 2 a 1**
- b. dos buffers triestado
- c. tres demultiplexores
- d. tres conexiones directas al bus

24. Una posible codificación en microinstrucciones de la instrucción call X es:

- a. $\text{sp}=\text{sp}-1 ; \text{m}[\text{sp}]=\text{pc} ; \text{pc}=\text{pc}+1$
- b. $\text{sp}=\text{pc}-1 ; \text{m}[\text{sp}]=\text{pc} ; \text{pc}=x$
- c. $\text{pc}=x ; \text{sp}=\text{sp}-1 ; \text{m}[\text{sp}]=\text{pc}$
- d. **$\text{sp}=\text{sp}-1 ; \text{m}[\text{sp}]=\text{pc} ; \text{pc}=x$**

25. En la secuencia de instrucciones siguiente, siendo el primer registro el destino, ¿cuántos riesgos se dan?

```
sub r4,r1,r3
or r8,r6,r4
```

- a. Un riesgo estructural
- b. **Un riesgo por dependencia de datos**
- c. Un riesgo estructural y dos por dependencia de datos
- d. Dos riesgos por dependencia de datos y uno de control

26. Un procesador x86 a 4 GHz dispone de 7 unidades de ejecución en paralelo, con 20 etapas de segmentación, y es capaz de emitir (comenzar a ejecutar) 4 instrucciones en cada ciclo de reloj. ¿Qué velocidad aproximada de ejecución de instrucciones será capaz de alcanzar (MIPS = millones de instrucciones por segundo)?

- a. 28000 MIPS
 - b. 16000 MIPS**
 - c. 4000 MIPS
 - d. 80000 MIPS
-

27. Se dispone de un procesador con una frecuencia de reloj de 1 GHz. Se le conecta un dispositivo que genera 100 000 interrupciones por segundo. La rutina de servicio de interrupción ejecuta 500 instrucciones. El número medio de ciclos por instrucción es 2. ¿Qué porcentaje del tiempo dedica el procesador al dispositivo?

- a. 1%
 - b. 10%**
 - c. 50%
 - d. 90%
-

28. La técnica de sondeo, escrutinio o "polling"...

- a. Se utiliza para identificar el destino de una interrupción.
 - b. Permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos.**
 - c. En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción.
 - d. Es incompatible con la técnica de daisy-chain.
-

29. ¿Cuántos conjuntos tendría una cache de 256 KB asociativa por conjuntos de 16 vías con líneas de 64 bytes?

- a. 4
 - b. 16
 - c. 64
 - d. 256**
-

30. ¿Qué componentes necesitamos para construir una memoria de 2K x 8 bits?

- a. 64 memorias de 128 x 1 bits y un decodificador de 4 a 16
 - b. 8 memorias de 512 x 2 bits y un decodificador de 1 a 2
 - c. 8 memorias de 256 x 4 bits y un decodificador de 2 a 4
 - d. 16 memorias de 512 x 2 bits y un decodificador de 2 a 4**
-

1. ¿De qué depende el tamaño del contador de programa?
 a) De la longitud del código de operación de las instrucciones.
 b) Del ancho del bus de datos.
 c) Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.
 d) **Del número de direcciones de memoria.**
2. Para direccionar una memoria de bytes en la que quepan 2G palabras de 32 bits se necesitarán:
 a) 31 bits como mínimo
 b) 32 bits exactamente
 c) **33 bits como mínimo**
 d) 21 bits como máximo
3. ¿De qué depende el tamaño del contador de programa?
 a) de la longitud del código de operación
 b) el tamaño no importa
 c) del ancho del bus de datos
 d) **ninguna de las otras respuestas es cierta**
4. Una computadora puede funcionar prescindiendo de:
 a) del PC
 b) **de un acumulador**
 c) del IR
 d) de los biestables de condición
5. Si usamos una estructura de bus con DMA:
 a) **la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.**
 b) al bus del sistema sólo se conecta la CPU y la MP, ya que el DMA se conecta directamente a MP para realizar las transferencias de datos.
 c) la velocidad de este controlador establece la velocidad del bus del sistema.
 d) podemos prescindir de controladores de E/S ya que el controlador de DMA se ocupa de controlar las transferencias hacia/desde los periféricos.
6. Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos y un computador Y tarda 14 segundos en ejecutar ese mismo programa, ¿cuántas veces es más rápido el computador Y que el X?
 a) 12
 b) 0,538
 c) **1,857**
 d) 1,538
7. ¿Cuál de las siguientes afirmaciones es incorrecta?

- a) En lenguaje ensamblador cada instrucción se corresponde con una instrucción máquina.
 - b) Un programa de alto nivel interpretado es más lento que el mismo programa en lenguaje máquina.
 - c) El lenguaje de alto nivel es más portable que el lenguaje máquina.
 - d) **En lenguaje ensamblador las instrucciones se escriben en binario.**
8. ¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?
- a) posición 0: FF pos.1:FF pos.2: FF pos.3: 80
 - b) **0:80 1:FF 2:FF 3:FF**
 - c) 0:00 1:01 2: 00 3:80
 - d) Ninguna de las anteriores
9. En las últimas generaciones de computadores la mejora de prestaciones viene dada por:
- a) avances en la estructura y arquitectura del computador.
 - b) **avances en tecnología y avances en la estructura y arquitectura del computador.**
 - c) avances en las tecnologías de fabricación.
 - d) avances en los sistemas operativos y aplicaciones.
10. ¿Qué modelo de programa se ejecuta en las arquitecturas Von Neumann?
- a) Externo.
 - b) Microprogramado.
 - c) Cableado.
 - d) **Almacenado**

Si queremos almacenar la palabra de 16 bits 0x9660 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x1000 como:

- a) M[0x1000]=0x06 y M[0x1001]=0x69
- b) M[0x1000]=0x69 y M[0x1001]=0x06
- c) M[0x1000]=0x96 y M[0x1001]=0x60
- d) M[0x1000]=0x60 y M[0x1001]=0x96

El computador EDVAC, propuesto por John Von Neumann, presentaba dos importantes diferencias respecto al ENIAC:

- a) **Empleaba aritmética binaria y permitía trabajar con un programa almacenado.**
- b) Utilizaba transistores y memoria de semiconductor.
- c) Se programaba enchufando centenares de clavijas y empleaba aritmética octal.
- d) Era electromecánico y de propósito específico.

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) El modo de direccionamiento permite determinar un operando o la ubicación del operando.
- b) Los operandos siempre están almacenados en memoria.
- c) El repertorio de instrucciones es el conjunto de operaciones que es capaz de interpretar la unidad de control.
- d) El repertorio de instrucciones debe ser capaz de realizar una tarea en un tiempo finito.

El registro MBR...

- a) especifica la dirección en memoria de la palabra que va a ser escrita o leída
- b) contiene el código de operación de la instrucción que se está ejecutando
- c) contiene la dirección de la próxima instrucción que va a ser captada de memoria
- d) contiene el valor que va a ser almacenado en la memoria, o bien se usa para recibir un valor procedente de la memoria

Una dirección de memoria se refiere siempre a:

- a) una palabra
- b) 16 bits
- c) un byte
- d) ninguna de las anteriores

En las instrucciones aritméticas con dos operandos de un procesador con arquitectura de pila, los dos operandos...

- a) son dos registros del procesador.
- b) se introducen en la pila tras realizar la operación.
- c) son la cima de la pila y el elemento siguiente de la cima de la pila.
- d) pueden estar en cualquier posición de la pila.

En un sistema con dos buses separados, uno para el subsistema de memoria y otro para la E/S...

- a) el bus que une la memoria y el procesador suele funcionar a la velocidad de la memoria
- b) el bus de E/S funciona a la velocidad del periférico más rápido
- c) ambos buses tienen que tener el mismo ancho de banda
- d) Ninguna de las respuesta anteriores es cierta

La idea de desarrollar máquinas CISC surgió para:

- a) simplificar el diseño hardware de la UC.
- b) tener instrucciones cercanas al lenguaje de alto nivel.
- c) conseguir un conjunto de instrucciones cortas y sencillas de decodificar.
- d) ninguna de las respuestas anteriores es cierta.

Un modo de vídeo de 512 x 256 píxeles y 16 colores por píxel ocupa una memoria de:

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- Perfecto para largas noches de risas con amigos



- a) 8 KB
 b) **64 KB**
 c) 2 MB
 d) 128 KB

Un computador con 8 bits en el bus de direcciones puede direccionar como máximo:

- a) 8192 palabras
 b) 16384 palabras
 c) **256 palabras**
 d) 1024 palabras

¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:

push #4

push #7

add

push #10

sub

- a) 11, 1
 b) **1**
 c) 4, 7, 10
 d) 4, 11, 1

Para direccionar una memoria de bytes en la que quepan 1G palabras de 32 bits se necesitarán:

- a) **32 bits**
 b) 33 bits
 c) 21 bits
 d) 31 bits

¿En qué generación, dentro de la historia de los computadores digitales, aparece la microprogramación?

- a) primera
 b) segunda
 c) **tercera**
 d) cuarta

Un computador que utilice el sistema big-endian, almacena el número 0x2143 a partir de la dirección 0 como:

- a) M[0]=0x43 y M[1]=0x21
 b) **M[0]=0x21 y M[1]=0x43**
 c) M[0]=0x34 y M[1]=0x12

- d) $M[0]=0x12$ y $M[1]=0x34$

Indique cuál de las siguientes afirmaciones sobre el ENIAC no es correcta:

- a) Un inconveniente era que tenía que ser programado manualmente mediante conmutadores, y conectando y desconectando cables.
- b) La idea del programa almacenado surgió durante el desarrollo del ENIAC, pero no fue implementada en este computador.
- c) Contaba con más de 17000 tubos de vacío.
- d) Como los computadores actuales, era una máquina binaria, es decir, los números estaban representados en forma binaria y los cálculos aritméticos se realizaban también en el sistema binario.

En una estructura de computador de bus único (bus del sistema):

- a) es la estructura más usada en los PC actuales
- b) la UC concede el acceso al bus, por lo que éste funciona a la velocidad de la CPU
- c) sólo una unidad funcional puede tener el control del bus en cada momento
- d) ninguna de las otras respuestas es cierta

El objetivo de un diseño CISC es...

- a) disminuir el tamaño medio de instrucción.
- b) disminuir el número de instrucciones a ejecutar por un programa.
- c) disminuir el número medio de ciclos por instrucción.
- d) disminuir la frecuencia de reloj.

¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?

- a) El resultado de la ejecución de un conjunto de programas de prueba.
- b) La arquitectura del procesador.
- c) El número medio de ciclos de reloj por instrucción.
- d) La frecuencia de reloj del procesador.

¿En qué generación, dentro de la historia de los computadores digitales, aparecen los sistemas operativos multiusuario?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

En el arbitraje de un bus...

- a) los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
- b) si hay un único dispositivo pasivo, siempre funciona como esclavo
- c) si hay varios dispositivos activos, siempre funcionan como maestros

- d) todas las respuestas anteriores son ciertas

¿Cuál de las siguientes características es posterior a la segunda generación de computadores?

- a) RISC.
- b) Memoria de núcleos de ferrita.
- c) Transistor.
- d) Lenguaje ensamblador.

¿Cuál de las siguientes no es una característica de los computadores RISC?

- a) La decodificación de las instrucciones debe ser simple: un computador RISC debería emplear un único formato de instrucción
- b) Un computador RISC no debe emplear microprogramación.
- c) Para acelerar el computador RISC se emplean técnicas de pipelining.
- d) Las funciones que realizan los computadores RISC deben ser lo más complejas y potentes que sea posible.

¿En qué generación, dentro de la historia de los computadores digitales, se alcanzaron tiempos de conmutación del orden de nanosegundos?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) Todas las instrucciones deben tener código de operación
- b) El formato de una instrucción nos indica el significado de cada bit de la instrucción
- c) Todas las instrucciones deben tener operando fuente y operando destino
- d) No siempre es necesario indicar la dirección de la siguiente instrucción

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "big-endian", quedará almacenada a partir de la posición 0x1000 como:

- a) M[0x1000]=0xA6 y M[0x1001]=0x91
- b) M[0x1000]=0x65 y M[0x1001]=0x89
- c) M[0x1000]=0x91 y M[0x1001]=0xA6
- d) M[0x1000]=0x89 y M[0x1001]=0x65

En el contexto del lenguaje máquina, el acrónimo ISA suele referirse a:

- a) Information Security Architecture
- b) Instruction Set Architecture
- c) Intel Standard Architecture

- d) Industry Standard Architecture

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x8600 como:

- a) $M[0x8600]=0x69$ y $M[0x8601]=0x85$
- b) $M[0x8600]=0x65$ y $M[0x8601]=0x89$
- c) $M[0x8600]=0x89$ y $M[0x8601]=0x65$
- d) $M[0x8600]=0x85$ y $M[0x8601]=0x69$

¿Qué tipo de instrucciones se emplean más en una arquitectura de acumulador?

- a) aritmético-lógicas
- b) de transferencia de datos con memoria
- c) de desplazamiento y rotación
- d) de transferencia de datos entre registros

En la arquitectura Von Neumann...

- a) el programa se encuentra residente en memoria.
- b) Todas son ciertas.
- c) los bloques principales son la unidad de control, la ALU y la CPU.
- d) los registros se encuentran en la memoria principal.

¿Cuál de las siguientes afirmaciones es falsa?

- a) la anchura del bus de datos es de 16 bits
- b) el bus de direcciones es unidireccional
- c) el bus de control transporta señales de estado
- d) el bus de datos es bidireccional

En las arquitecturas RISC hay...

- a) muchos registros y pocos modos de direccionamiento.
- b) pocos modos de direccionamiento y muchos formatos de instrucción.
- c) pocos registros y muchos tipos de instrucciones.
- d) pocas instrucciones muy rápidas con muchos modos de direccionamiento.

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria virtual?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

No en todas las instrucciones máquina hay una fase de

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- Perfecto para largas noches de risas con amigos



- a) decodificación
- b) ejecución
- c) captación
- d) **captura de operandos**

¿Cuál de las siguientes afirmaciones acerca de la arquitectura Von Neumann es cierta?

- a) La arquitectura Von Neumann es un diseño que sitúa el programa en un almacenamiento distinto al de los datos.
- b) Para ejecutar un programa en una máquina Von Neumann, es necesario volver a cablear o incluso rediseñar la máquina.
- c) **La separación entre almacenamiento y unidad de procesamiento es una de los ideas contempladas en la arquitectura Von Neumann.**
- d) Existe un consenso general en considerar justo el término "arquitectura von Neumann", ya que las ideas de esta arquitectura fueron completamente originales de John Von Neumann y no influenciadas por sus colaboradores o predecesores.

¿Qué dice la ley de Moore?

- a) Todas las respuestas son ciertas.
- b) Que el tamaño de los transistores se duplica cada 18 meses.
- c) **Que el número de transistores de un chip se duplica cada 18 meses.**
- d) Que la memoria de los ordenadores se duplica cada 18 meses.

Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro

- a) Contador de programa
- b) **De instrucción**
- c) Puntero de pila
- d) De direcciones

Una máquina superescalar es aquella que:

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- c) **emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.**
- d) ninguna respuesta de las anteriores es correcta.

¿Qué modelo de programa se ejecuta en las arquitecturas de Von Neumann?

- | | |
|-----------------------|----------------------|
| <input type="radio"/> | a) microprogramado |
| <input type="radio"/> | b) nanoprogramado |
| <input type="radio"/> | c) almacenado |
| <input type="radio"/> | d) cableado |

Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria. Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de 2^{10} , ¿qué tamaño de memoria admite el computador?

- a) 2^{10} palabras
- b) **2^{12} palabras**
- c) 2^{13} palabras
- d) Ninguna de las anteriores

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

- a) registro-registro
- b) registro-memoria
- c) **memoria-memoria**
- d) ninguna de las anteriores es cierta

En la captación de un operando que reside en memoria:

- a) en MBR indicamos la dirección donde está y en MAR lo recogemos
- b) en MBR indicamos la dirección donde está y en IR lo recogemos
- c) **en MAR indicamos la dirección donde está y en MBR lo recogemos**
- d) en MAR indicamos la dirección donde está y en IR lo recogemos

En el direccionamiento inmediato, tras captarse completamente la instrucción:

- a) se accede al operando, que está contenido en una posición de memoria principal.
- b) se accede al operando, que se encuentra almacenado en uno de los registros programables.
- c) **se accede al operando, que es una constante contenida en la propia instrucción.**
- d) el código de operación contiene el operando.

El espacio direccionable de memoria de un computador depende del diseño del:

- a) **Bus de direcciones**
- b) Bus de datos
- c) a) y b) son correctas
- d) Ninguna de las anteriores es correcta

La primera generación de computadores se caracteriza por el uso de:

- a) Microprocesadores.
- b) Fibra óptica.
- c) Transistores.
- d) **Tubos de vacío.**

¿En qué registro está contenido el último dato (o instrucción) leído de memoria, o el dato que se va a escribir en memoria?

- a) MAR.
- b) **MBR.**
- c) Acumulador.
- d) PC.

El ancho de palabra de una memoria corresponde a:

- a) La longitud del registro de direcciones de la memoria.
- b) El número de posiciones que la componen.
- c) El número que identifica únicamente cada posición de la memoria.
- d) **La cantidad de bits que caben en una sola posición**

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) **El tamaño de una instrucción en lenguaje máquina siempre ocupa dos bytes en los procesadores RISC.**
- b) En las arquitecturas CISC hay más instrucciones que en las RISC.
- c) Las arquitecturas RISC son del tipo registro-registro.
- d) Las arquitecturas RISC simplifican la decodificación.

¿Qué medida de prestaciones es la más fiable de todas las posibles?

- a) **ninguna de las otras respuestas es correcta**
- b) MIPS equivalentes
- c) MFLOPS
- d) MIPS

¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop?:

```
push #1  
push #2  
push #3  
pop a  
push #4  
pop a  
pop a
```

- a) 1 y 2
- b) 10

- c) 1
- d) 1, 2, 3 y 4

El programador de lenguaje ensamblador necesita conocer:

- a) el diseño RTL del procesador.
- b) todas las respuestas son ciertas.
- c) la arquitectura del ordenador.
- d) la microarquitectura del procesador.

¿En qué generación, dentro de la historia de los computadores digitales, aparecen las memorias de semiconductores?

- a) Segunda generación.
- b) Tercera generación.
- c) Cuarta generación.
- d) Quinta generación.

El direccionamiento directo a memoria utiliza...

- a) Dos desplazamientos contenidos en la misma instrucción.
- b) Un registro y un desplazamiento contenidos en la propia instrucción.
- c) Un desplazamiento.
- d) Un registro.

El ancho de palabra de una memoria corresponde a:

La longitud del registro de datos de la memoria.

Siendo EDX=0xf000 y ECX=0x100 ¿Cuál de las siguientes instrucciones tiene como dirección efectiva 0xf400?

xorl (%edx,%ecx,4), %eax.

Para direccionar una memoria de 1 G palabras de 32 bits cada una, que se direcciona byte a byte, se necesitarán:

32bits.

Un modo de vídeo de 1680 x 1050 píxeles y 4 canales de color por píxel (R, G, B, alpha) y 256 niveles de intensidad por canal (0 a 255), ocupa aproximadamente una memoria de:

- a) 7 MB
- b) 1,7 GB
- c) 54 MB
- d) 1,7 MB

Suponiendo que todos los registros inicialmente contienen el valor 1, ¿cuál es el valor de ECX tras la ejecución de la siguiente secuencia de instrucciones?

- **480 cartas resultado de mucho amor y birra**
- **80 cartas especiales para animar tus fiestas**
- **A partir de 16 años, de 3 a 15 jugadores**
- Perfecto para largas noches de risas con amigos



```
mov $4, %eax
mov $3, %ebx
lea (%eax, %eax), %ecx
sub %ebx, %ecx
imul %ecx, %eax
```

- a) 6
- b) 36
- c) 20
- d) 5

En una arquitectura de registros (a nivel de lenguaje máquina):

- a) operar usando registros es más rápido.
- b) la generación de código resulta más simple que en arquitecturas de pila o acumulador.
- c) tiene el problema de que las instrucciones pueden ser largas.
- d) **todas las respuestas anteriores son ciertas.**

En las arquitecturas RISC:

- a) **se usan instrucciones muy simples que se pueden segmentar.**
- b) la programación resulta más simple que en los CISC.
- c) la UC suele ser más compleja que en CISC, al ser cableada.
- d) realmente se usan pocas instrucciones de las disponibles en el conjunto de instrucciones.

En una máquina con 32 registros direccionables e instrucciones de 16 bits es posible codificar:

- a) 64 instrucciones de dos registros y 32 instrucciones de un registro.
- b) 2 instrucción de tres registros y 32 instrucciones de dos registros.
- c) **1 instrucción de tres registros, 31 instrucciones de dos registros, 31 instrucciones de un registro y 32 instrucciones de 0 direcciones.**
- d) 63 instrucciones de dos registros, 32 instrucciones de un registro y 32 instrucciones de 0 direcciones.

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

- a) **SS * 10h + SP**
- b) SS * 10h + BP
- c) SP
- d) BP

Sobre el direccionamiento relativo a contador de programa:

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos y llamadas a subrutinas reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está

ejecutando.

- d) Todas las respuestas son ciertas.

En el RISC-I, una ventana de registros contiene, entre otros registros,...

- a) **registros para recibir parámetros del procedimiento llamador**
- b) registros para almacenar matrices de enteros
- c) registros para enviar parámetros a otros procesos
- d) todas las respuestas son falsas

¿Cuál de las siguientes afirmaciones es cierta?

- a) **El modo de direccionamiento indirecto a través de registro es el modo más sencillo de direccionamiento a memoria.**
- b) Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.
- c) En el direccionamiento inmediato, la dirección efectiva se encuentra en un registro de uso general del procesador.
- d) En el direccionamiento a registro, el objeto direccionado es una constante contenida en la propia instrucción.

Una instrucción de lenguaje máquina debe:

- a) tener un número fijo de operandos, ya sean implícitos o explícitos.
- b) realizar una tarea sencilla y única, independiente de otras instrucciones.
- c) ser autocontenido e independiente de otras instrucciones o de su posición en el programa.
- d) **todas las respuestas anteriores son ciertas**

¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción mov bx, 5h?

- a) Direccionamiento inmediato
- b) Direccionamiento implícito
- c) **Direccionamiento directo a registro**
- d) Direccionamiento relativo a registro base

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

- a) **JNE dir**
- b) JMP dir
- c) CALL dir
- d) RET

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

- a) **directo a registro**
- b) a registro inmediato

- c) a registro literal
- d) ninguno de los anteriores

¿Cuál de las siguientes parejas de microprocesadores representa mejor los conceptos RISC?

- a) Itanium, Alpha
- b) Pentium, Athlon
- c) PA-RISC, PowerPC
- d) **MIPS, SPARC**

Si AX = 0xFA50 y ejecutamos AND \$0xFF, %AX

- a) El registro AH se pone a FF
- b) El registro AL se pone a 0
- c) El registro AL se pone a FF
- d) **El registro AH se pone a 0**

Las arquitecturas de acumulador se caracterizan por:

- a) facilidad de generación de código eficiente para los compiladores.
- b) **diseño simple del procesador.**
- c) instrucciones muy largas.
- d) tráfico reducido entre procesador y memoria.

¿Cuál de las siguientes afirmaciones sobre compilación C->ASM es falsa?

- a) Puede que el compilador altere el orden del código C, apareciendo antes la traducción de sentencias C posteriores
- b) **Puede que el compilador altere por optimización el orden de los parámetros pasados en una llamada a función, si el marco de pila resultante es más eficiente**
- c) Puede suceder que varias sentencias C se traduzcan por una única instrucción ASM
- d) Puede que el compilador elimine por optimización construcciones C enteras (como un bucle for completo), si se conoce el resultado en tiempo de compilación

La instrucción setg %al:

- a) No cambia el contenido de AL
- b) Pone siempre AL a 1.
- c) Complementa AL si el resultado de la comparación anterior es A > B.
- d) **Pone AL a 1 en algunos casos.**

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

- a) 4C000h

- b) 53000h
- c) 5C000h
- d) 15025h

Si queremos almacenar la palabra de 16 bits 8965h en memoria según "big-endian", quedará almacenada a partir de la posición 1000h como:

- a) en el byte 1000h se guarda 91h y en el 1001h A6h
- b) en el byte 1000h se guarda 65h y en el 1001h 89h
- c) en el byte 1000h se guarda 89h y en el 1001h 65h
- d) en el byte 1000h se guarda A6h y en el 1001h 91h

¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción mov bx, 5h?

- a) Direccionamiento relativo a registro base
- b) Direccionamiento implícito
- c) Direccionamiento inmediato
- d) Direccionamiento directo a registro

Si %eax contiene x, ¿cuál de las siguientes instrucciones calcula x*5?

- a) leal (%eax,%eax,4),%edx
- b) leal 3(%eax,%eax,2),%edx
- c) leal 4(%eax,%eax),%edx
- d) leal 3(%eax,2),%edx

En el RISC-I, una ventana de registros contiene:

- a) registros para recibir parámetros del procedimiento llamador.
- b) registros para almacenar variables locales.
- c) registros para enviar parámetros a procedimientos.
- d) todas las respuestas son ciertas.

El microprocesador de la familia x86 usado en los primeros IBM PC originales fue el:

- a) 8088
- b) Pentium
- c) 8085
- d) 80486

Sobre el ensamblador:

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.

¿BUSCAS LIBROS O EBOOKS SOBRE ENFERMERÍA?

AXON librería especializada en Ciencias de la Salud.

AXON
axon.es



Encuentra todos los libros y eBooks para tu especialización, además del material complementario que necesites.

- d) Todas las respuestas son ciertas

¿Cuál de las siguientes listas está correctamente ordenada temporalmente?

- a) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
- b) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
- c) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
- d) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX

En el direccionamiento indirecto a través de registro, la dirección efectiva...

- a) se encuentra en el registro de instrucción.
- b) se calcula como la suma del contenido de dos registros.
- c) se encuentra en una dirección de memoria.
- d) se encuentra en un registro general del procesador.

Una instrucción de salto si igual tiene que comprobar el valor de:

- a) el bit de signo
- b) el bit de acarreo
- c) el bit de cero
- d) los bits de signo y desbordamiento

En una máquina de 32 registros direccionables e instrucciones de 16 bits:

- a) No se pueden codificar a la vez 63 instrucciones de dos registros, una instrucción de 0 direcciones y 32 instrucciones de 1 registro.
- b) No se pueden codificar a la vez 62 instrucciones de dos registros, 32 instrucciones de 1 registro y 64 instrucciones de 0 direcciones.
- c) No se pueden codificar a la vez 64 instrucciones de 1 registro, 32 instrucciones de 2 registros y 32 instrucciones de 0 direcciones.
- d) No se pueden codificar a la vez 64 instrucciones de 0 direcciones, 63 instrucciones de 2 registros y 16 instrucciones de 1 registro.

EXAMEN FALIN

- 1 En la práctica "media" un estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

Elec
ción
único

bucle:

a

```
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %edx, %ebp
    jnc nocarry
    inc %edx
```

nocarry:

```
    inc %esi
    cmp %esi,%ecx
    jne bucle
```

Estando bien programado todo lo demás, este código...

P
U
r
o
s
f
u
e
a
s
r
i
o
r
o
e
s

fallaría con lista: .int -1,-2,-4,-8

)

) no siempre produce el
) resultado correcto, pero el error
no se manifiesta en los

ejemplos propuestos, o se manifiesta en ambos

- ✓ • c produce siempre el resultado correcto
c fallaría con lista: .int 0,1,2,3

Puntuación: **1,00**

¿Cuál de las siguientes funciones en C NO puede traducirse al siguiente código en ensamblador?

```
movl %esi, %esi
movq (%rdi,%rsi,8), %rax
ret
```

Usuar Profesor

io es

- a) unsigned long *elem (unsigned long *v[20],unsigned i)
{
 return v[i];
}
- b) unsigned long elem (unsigned long v[20], unsigned i) {
 return v[i];
}
- ✓ • c) unsigned long elem (unsigned long v[20],unsigned *i) {
 return v[*i];
 }
- d) unsigned *elem (unsigned *v[20],unsigned i) {
 return v[i];
}

Puntuación: **1,00**

¿Qué valor contendrá %edx tras ejecutar las siguientes instrucciones?

```
xor %eax, %eax
sub $1, %eax
cltd
idiv %eax
```

Usuar Profesor

io es

- ✓ • a) 0

- b) 1
- c) no puede saberse con los datos del enunciado
- d) -1

Puntuación: **1,00**

¿Cuál de los siguientes grupos de instrucciones sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

Usuar Profesor

io es

- a) CMP, TEST
- b) AND, OR,
XOR
- c) IMUL, IDIV
- d) ADC, SBB

Puntuación: **1,00**

En x86-64, es responsabilidad del procedimiento llamado (callee) guardar, entre otros, los registros:

Usuar Profesor

io es

- a) %r12, %r13
- b) %rbx, %rsi, %rdi
- c) %rax, %rbx, %rcx, %rdx
- d) %rax, %r10, %r11

Puntuación: **1,00**

La instrucción movq %rbp,%rsp

Usuar Profesor

io es

- a) Introduce en la pila el contenido del registro RBP.
- b) Copia el contenido del registro RBP en el registro RSP.
- c) Copia el contenido del registro RSP en el registro RBP.
- d) Introduce en la pila el contenido del registro RSP.

Puntuación: **1,00**

¿Cuál de los siguientes registros no forma parte del registro de 64 bits RAX?

Usuar Profesor

io es

- a) AX
- b) AH
- c) EA
X
- ✓ • d) DA
X

Puntuación: **1,00**

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x^5 ?

Usuar Profesor

io es

- a) leaq 3(%rax,%rax,2),%rdx
- b) leaq 3(,%rax,2),%rdx
- ✓ • c) leaq (%rax,%rax,4),%rdx
- d) leaq 4(%rax,%rax),%rdx

Puntuación: **1,00**

Si RCX vale 0, la instrucción adc \$-1,%rcx

Usuar Profesor

io es

- a) Pone CF=0 (independientemente de lo que valiera antes)
- b) Pone CF=1 (independientemente de lo que valiera antes)
- c) No cambia CF (si valía 0 permanecerá a 0, si valía 1 permanecerá a 1)
- X d) Cambia CF (si valía 0 cambiará a 1, si valía 1 cambiará a 0)

Puntuación: **-0,33**

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

Usuar Profesor

io es

- a) JMP dir
- ✓ • b) JNE dir
- c) CALL
dir
- d) RET

Puntuación: **1,00**

Tras dividir 0x00000000 00040000 (en EDX:EAX) entre 0x00008000 (ECX), el resultado será:

Usuar Profesor

io es

- a) 0x0 en EAX y 0x8 en EDX
- b) 0x8 en AX y 0x8 en DX
- c) 0x8 en EAX y 0x0 en EDX
- d) 0x0 en AX y 0x8 en DX

Puntuación: **1,00**

Si desplazamos RBX mediante salq \$3, %rbx:

Usuar Profesor

io es

- a) Lo multiplicamos por 8
- b) Lo dividimos por 8
- c) Lo multiplicamos por 4
- d) Lo dividimos por 4

Puntuación: **1,00**

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x*9?

Usuar Profesor

io es

- a) leaq (%rax,%rax,8),%rdx
- b) leaq 8(%rax,%rax),%rdx
- c) leaq 5(%rax,%rax,4),%rdx
- d) leaq 3(%rax,%rax,2),%rdx

Puntuación: **-0,33**

Dada la siguiente declaración:

longsaludo: .quad . - saludo

¿Cuál de las siguientes órdenes de gdb NO muestra el contenido de longsaludo?

Usuar Profesor

io es

- a) x/1dg &longsaludo
- b) p *((long *)

&longsaludo)

- c) p (long) &longsaludo
- d) x/8db &longsaludo

Puntuación: **-0,33**

En la práctica “media” se pide sumar una lista de 16 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista de 16 enteros causaría overflow con 32bits?

PISTA: Sumar un número 16 veces == multiplicarlo por 16 == desplazarlo 4 bits a la izquierda

Usuar Profesor

io es

- a) 0xffff ffff
- b) 0xf000
 0000
- ✓ • c) 0xf7ff ffff
- d) 0xfc00
 0000

Puntuación: **1,00**

¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar)

Usuar Profesor

io es

- a) movw %dx, (%eax)
- b) movswl (%eax), %edx
- ✓ • c) pushb \$0xFF
- d) movzbl %dl, %eax

Puntuación: **1,00**

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuar Profesor

io es

- a) No siempre es necesario indicar la dirección de la siguiente instrucción
- ✓ • b) Todas las instrucciones deben tener operando fuente y operando destino

- c) El formato de una instrucción nos indica el significado de cada bit de la instrucción
- d) Todas las instrucciones deben tener código de operación

Puntuación: **1,00**

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuar Profesor

io es



- a) El direccionamiento indirecto indica la dirección del operando.
- b) En el direccionamiento implícito no se indica la ubicación del operando
- c) El direccionamiento indexado es útil para manejo de vectores.
- d) En el direccionamiento inmediato el dato se encuentra en la propia instrucción

Puntuación: **1,00**

La línea de código ensamblador:

mov \$msg, %rsi

Usuar Profesor

io es



- a) Copia en RSI todo el contenido de la cadena apuntada por msg.
- b) Copia en RSI los primeros 64 bits de memoria desde la posición apuntada por la etiqueta msg.
- c) Copia en RSI los 64 bits de la dirección msg.
- d) Copia en RSI la dirección de memoria de 64 bits almacenada en memoria a partir de la posición indicada por la etiqueta msg.

Puntuación: **-0,33**

¿De qué depende el tamaño del contador de programa?

Usuar Profesor

io es



- a) Del número de direcciones de memoria.
- b) Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.
- c) De la longitud del código de operación de las instrucciones.
- d) Del ancho del bus de datos.

Puntuación: **1,00**

1

Elección única

Si RCX vale 0, la instrucción adc \$-1,%rcx

Usuario Profesores

a)

Pone CF=0 (independientemente de lo que valiera antes)

b)

Cambia CF (si valía 0 cambiará a 1, si valía 1 cambiará a 0)

c)

Pone CF=1 (independientemente de lo que valiera antes)

• d)

No cambia CF (si valía 0 permanecerá a 0, si valía 1 permanecerá a 1)

Puntuación: -0,33

2

Elección única

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x^5 ?

Usuario Profesores

• a)

leaq (%rax,%rax,4),%rdx

b)

leaq 3(%rax,2),%rdx

c)

leaq 3(%rax,%rax,2),%rdx

d)

leaq 4(%rax,%rax),%rdx

Puntuación: 1,00

3

Elección única

¿Cuál de los siguientes registros no forma parte del registro de 64 bits RAX?

Usuario Profesores

a)

EAX

b)

AX

• c)

DAX

d)

AH

Puntuación: 1,00

4

Elección única

¿De qué depende el tamaño del contador de programa?

Usuario Profesores

- ✓ • a)

Del número de direcciones de memoria.

b)

Del ancho del bus de datos.

c)

De la longitud del código de operación de las instrucciones.

d)

Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.

Puntuación: 1,00

5

Elección única

En la práctica "media" un estudiante usa el siguiente bucle para acumular la suma en EBP:EDI antes de calcular la media y el resto

bucle:

```
    mov (%ebx,%esi,4), %eax
    cltd
    add %eax, %edi
    adc %edx, %ebp
    jnc nocarry
    inc %edx
```

nocarry:

```
    inc %esi
    cmp %esi,%ecx
    jne bucle
```

Estando bien programado todo lo demás, este código...

Usuario Profesores

a)

fallaría con lista: .int 0,1,2,3

- ✓ • b)

produce siempre el resultado correcto

c)

no siempre produce el resultado correcto, pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos

d)

fallaría con lista: .int -1,-2,-4,-8

Puntuación: 1,00

6

Elección única

Dada la siguiente declaración:

longsaludo: .quad . - saludo

¿Cuál de las siguientes órdenes de gdb NO muestra el contenido de longsaludo?

Usuario Profesores

a)

p *((long *) &longsaludo)

b)

x/1dg &longsaludo

c)

x/8db &longsaludo

d)

p (long) &longsaludo

Puntuación: -0,33

7

Elección única

¿Cuál de los siguientes grupos de instrucciones sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

Usuario Profesores

a)

IMUL, IDIV

b)

ADC, SBB

c)

CMP, TEST

d)

AND, OR, XOR

Puntuación: 1,00

8

Elección única

¿Cuál de las siguientes afirmaciones es cierta?

Usuario Profesores

a)

En el modo de direccionamiento implícito, el objeto direccionado es una constante contenida en la propia instrucción.

b)

Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.

c)

En el direccionamiento directo, la dirección efectiva se encuentra en un registro de uso general del procesador.

- ✓ • d)

El direccionamiento a registro es similar al directo. La diferencia es que el campo de direcciones referencia un registro en lugar de una dirección de memoria principal.

Puntuación: 1,00

9

Elección única

En la práctica "media" se programa la suma de una lista de 16 enteros de 4 B para producir un resultado de 8 B, primero sin signo y luego con signo. Si la lista se rellena con el valor 0x0400 0000, ¿en qué se diferencian los resultados de ambos programas?

Usuario Profesores

- a)

en uno ocupa 32 bits, en otro 64 bits

- b)

no se diferencian

- c)

en uno se interpreta como negativo, en otro como positivo

- d)

en uno los 32 bits superiores son 0xFFFF FFFF, en el otro no

Puntuación: -0,33

10

Elección única

La línea de código ensamblador:

mov \$msg, %rsi

Usuario Profesores

- a)

Copia en RSI los primeros 64 bits de memoria desde la posición apuntada por la etiqueta msg.

- b)

Copia en RSI todo el contenido de la cadena apuntada por msg.

- c)

Copia en RSI los 64 bits de la dirección msg.

- d)

Copia en RSI la dirección de memoria de 64 bits almacenada en memoria a partir de la posición indicada por la etiqueta msg.

Puntuación: -0,33

11

Elección única

Si desplazamos RBX mediante salq \$3, %rbx:

Usuario Profesores

a)

Lo dividimos por 8

b)

Lo dividimos por 4

c)

Lo multiplicamos por 4

✓ • d)

Lo multiplicamos por 8

Puntuación: 1,00

12

Elección única

En la práctica “media” se pide sumar una lista de 16 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista de 16 enteros causaría overflow con 32bits?

PISTA: Sumar un número 16 veces == multiplicarlo por 16 == desplazarlo 4 bits a la izquierda

Usuario Profesores

a)

0xf000 0000

b)

0xffff ffff

c)

0xfc00 0000

✓ • d)

0xf7ff ffff

Puntuación: 1,00

13

Elección única

¿Cuál de las siguientes funciones en C NO puede traducirse al siguiente código en ensamblador?

```
movslq %esi, %rsi  
movq (%rdi,%rsi,8), %rax  
ret
```

Usuario Profesores

a)

```
unsigned *elem (unsigned *v[],int i) {  
    return v[i];  
}
```

b)
unsigned long *elem (unsigned long *v[],int i) {
 return v[i];
}

c)
unsigned long elem (unsigned long v[], int i) {
 return v[i];
}

✓ • d)
unsigned long elem (unsigned long v[],int *i) {
 return v[*i];
}

Puntuación: 1,00

14

Elección única

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x*9?

Usuario Profesores

- a)
leaq 8(%rax,%rax),%rdx
b)
leaq 3(%rax,%rax,2),%rdx
✓ • c)
leaq (%rax,%rax,8),%rdx
d)
leaq 5(%rax,%rax,4),%rdx

Puntuación: 1,00

15

Elección única

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuario Profesores

- a)
Todas las instrucciones deben tener código de operación
✓ • b)
Todas las instrucciones deben tener operando fuente y operando destino
c)
No siempre es necesario indicar la dirección de la siguiente instrucción
d)
El formato de una instrucción nos indica el significado de cada bit de la instrucción

Puntuación: 1,00

16

Elección única

¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar)

Usuario Profesores

✓ • a)

pushb \$0xFF

b)

movswl (%eax), %edx

c)

movw %dx, (%eax)

d)

movzbl %dl, %eax

Puntuación: 1,00

17

Elección única

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuario Profesores

a)

El direccionamiento indexado es útil para manejo de vectores.

✓ • b)

El direccionamiento indirecto indica la dirección del operando.

c)

En el direccionamiento implícito no se indica la ubicación del operando

d)

En el direccionamiento inmediato el dato se encuentra en la propia instrucción

Puntuación: 1,00

18

Elección única

¿Cuál de las siguientes funciones en C NO puede traducirse al siguiente código en ensamblador?

```
    movl %esi, %esi
    movq (%rdi,%rsi,8), %rax
    ret
```

Usuario Profesores

a)

```
unsigned *elem (unsigned *v[20],unsigned i) {
    return v[i];
```

}

b)

```
unsigned long *elem (unsigned long *v[20],unsigned i) {
```

```
    return v[i];
}
✓     •      c)
unsigned long elem (unsigned long v[20],unsigned *i) {
    return v[*i];
}
d)
unsigned long elem (unsigned long v[20], unsigned i) {
    return v[i];
}
Puntuación: 1,00
```

19

Elección única

Tras dividir 0x00000000 00040000 (en EDX:EAX) entre 0x00008000 (ECX), el resultado será:

Usuario Profesores
✓ • a)
0x8 en EAX y 0x0 en EDX
b)
0x0 en EAX y 0x8 en EDX
c)
0x8 en AX y 0x8 en DX
d)
0x0 en AX y 0x8 en DX

Puntuación: 1,00

20

Elección única

En la secuencia de programa siguiente:

```
5f5: e8 cd ff ff ff      callq  5c7 <suma>
5fa: 48 83 c4 20          add    $0x20,%rsp
```

¿cuál es el valor que introduce en la pila la instrucción callq?

Usuario Profesores
a)
0xffffffffcd
b)
0x5f5
c)
0x5c7
✓ • d)

0x5fa

Puntuación: 1,00

1

Elección única

Dada la siguiente declaración:

longsaludo: .quad . - saludo

¿Cuál de las siguientes órdenes de gdb NO muestra el contenido de longsaludo?

Usuario Profesores

• a)

p (long) &longsaludo

b)

x/8db &longsaludo

c)

x/1dg &longsaludo

X d)

p *((long *) &longsaludo)

Puntuación: -0,33

2

Elección única

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

Usuario Profesores

✓ • a)

JNE dir

b)

JMP dir

c)

RET

d)

CALL dir

Puntuación: 1,00

3

Elección única

Si desplazamos RBX mediante salq \$3, %rbx:

Usuario Profesores

a)

Lo dividimos por 8

✓ • b)

Lo multiplicamos por 8

c)

Lo multiplicamos por 4

d)

Lo dividimos por 4

Puntuación: 1,00

4

Elección única

¿Cuál de las siguientes afirmaciones es incorrecta?

Usuario Profesores

a)

Todas las instrucciones deben tener código de operación

b)

No siempre es necesario indicar la dirección de la siguiente instrucción

✓ • c)

Todas las instrucciones deben tener operando fuente y operando destino

d)

El formato de una instrucción nos indica el significado de cada bit de la instrucción

Puntuación: 1,00

5

Elección única

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x^9 ?

Usuario Profesores

✓ • a)

leaq (%rax,%rax,8),%rdx

b)

leaq 5(%rax,%rax,4),%rdx

c)

leaq 8(%rax,%rax),%rdx

d)

leaq 3(%rax,%rax,2),%rdx

Puntuación: 1,00

6

Elección única

¿Cuál de los siguientes registros no forma parte del registro de 64 bits RAX?

Usuario Profesores

a)

AH

✓ • b)

DAX

c)

AX

d)

EAX

Puntuación: 1,00

7

Elección única

¿Cuál de las siguientes líneas de código sirve para definir una variable entera de 64 bits llamada tam en GNU/as Linux x86-64?

Usuario Profesores

✓ • a)

 tam: .quad .-msg

b)

 _int tam = 0

c)

 int tam;

d)

 var tam : integer;

Puntuación: 1,00

8

Elección única

En la práctica “media” se pide sumar una lista de 16 enteros CON signo de 32 bits en una plataforma de 32 bits sin perder precisión, esto es, evitando overflow. ¿Cuál es el mayor valor negativo (menor en valor absoluto) que repetido en toda la lista de 16 enteros causaría overflow con 32bits?

PISTA: Sumar un número 16 veces == multiplicarlo por 16 == desplazarlo 4 bits a la izquierda

Usuario Profesores

a)

 0xf000 0000

✓ • b)

 0xf7ff ffff

c)

 0xfc00 0000

d)

 0xffff ffff

Puntuación: 1,00

9

Elección única

Tras dividir 0x00000000 00040000 (en EDX:EAX) entre 0x00008000 (ECX), el resultado será:

Usuario Profesores

a)

 0x0 en EAX y 0x8 en EDX

✓ • b)

 0x8 en EAX y 0x0 en EDX

c)

 0x8 en AX y 0x8 en DX

d)

0x0 en AX y 0x8 en DX

Puntuación: 1,00

10

Elección única

En la secuencia de programa siguiente:

5f5: e8 cd ff ff ff callq 5c7 <suma>

5fa: 48 83 c4 20 add \$0x20,%rsp

¿cuál es el valor que introduce en la pila la instrucción callq?

Usuario Profesores

a)

0xffffffffcd

b)

0x5c7

c)

0x5f5

✓ • d)

0x5fa

Puntuación: 1,00

11

Elección única

¿De qué depende el tamaño del contador de programa?

Usuario Profesores

✓ • a)

Del número de direcciones de memoria.

b)

Del ancho del bus de datos.

c)

Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.

d)

De la longitud del código de operación de las instrucciones.

Puntuación: 1,00

12

Elección única

En x86-64, el registro contador de programa se denomina:

Usuario Profesores

✓ • a)

RIP

b)

PCR

c)

PC

d)

EIP

Puntuación: 1,00

13

Elección única

La línea de código ensamblador:

mov \$msg, %rsi

Usuario Profesores

- a)

Copia en RSI los 64 bits de la dirección msg.

- b)

Copia en RSI todo el contenido de la cadena apuntada por msg.

- c)

Copia en RSI la dirección de memoria de 64 bits almacenada en memoria a partir de la posición indicada por la etiqueta msg.

- d)

Copia en RSI los primeros 64 bits de memoria desde la posición apuntada por la etiqueta msg.

Puntuación: -0,33

14

Elección única

¿Qué valor contendrá %edx tras ejecutar las siguientes instrucciones?

xor %eax, %eax

sub \$1, %eax

cltd

idiv %eax

Usuario Profesores

- a)

-1

- b)

0

- c)

no puede saberse con los datos del enunciado

- d)

1

Puntuación: 1,00

15

Elección única

La instrucción movq %rbp,%rsp

Usuario Profesores

- a)

Copia el contenido del registro RSP en el registro RBP.

b)

Introduce en la pila el contenido del registro RSP.

c)

Introduce en la pila el contenido del registro RBP.

✓ • d)

Copia el contenido del registro RBP en el registro RSP.

Puntuación: 1,00

16

Elección única

¿Cuál de las siguientes funciones en C NO puede traducirse al siguiente código en ensamblador?

```
movl %esi, %esi  
movq (%rdi,%rsi,8), %rax  
ret
```

Usuario Profesores

a)

```
unsigned long elem (unsigned long v[20], unsigned i) {  
    return v[i];  
}
```

b)

```
unsigned long *elem (unsigned long *v[20],unsigned i) {  
    return v[i];  
}
```

✓ • c)
unsigned long elem (unsigned long v[20],unsigned *i) {
 return v[*i];
}

d)

```
unsigned *elem (unsigned *v[20],unsigned i) {  
    return v[i];  
}
```

Puntuación: 1,00

17

Elección única

Si RAX contiene x, ¿cuál de las siguientes instrucciones calcula x*5?

Usuario Profesores

a)

leaq 3(%rax,%rax,2),%rdx

✓ • b)

leaq (%rax,%rax,4),%rdx

c)

leaq 4(%rax,%rax),%rdx

d)

leaq 3(%rax,2),%rdx

Puntuación: 1,00

18

Elección única

¿Cuál de los siguientes grupos de instrucciones sólo modifican los indicadores de estado sin almacenar el resultado de la operación?

Usuario Profesores

a)

IMUL, IDIV

b)

ADC, SBB

✓ • c)

CMP, TEST

d)

AND, OR, XOR

Puntuación: 1,00

19

Elección única

¿Cuál de las siguientes funciones en C NO puede traducirse al siguiente código en ensamblador?

```
movslq %esi, %rsi  
movq (%rdi,%rsi,8), %rax  
ret
```

Usuario Profesores

a)

```
unsigned *elem (unsigned *v[],int i) {  
    return v[i];  
}
```

✓ • b)

```
unsigned long elem (unsigned long v[],int *i) {  
    return v[*i];  
}
```

c)

```
unsigned long *elem (unsigned long *v[],int i) {  
    return v[i];  
}
```

d)

```
unsigned long elem (unsigned long v[], int i) {  
    return v[i];  
}
```

Puntuación: 1,00

20

Elección única

¿Cuál de las siguientes afirmaciones es cierta?

Usuario Profesores

a)

En el direccionamiento directo, la dirección efectiva se encuentra en un registro de uso general del procesador.

✓ • b)

El direccionamiento a registro es similar al directo. La diferencia es que el campo de direcciones referencia un registro en lugar de una dirección de memoria principal.

c)

Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.

d)

En el modo de direccionamiento implícito, el objeto direccionado es una constante contenida en la propia instrucción.

Puntuación: 1,00

El tamaño del registro indicador de una memoria asociativa de n palabras de m bits es...

Usuar Profesor

io es

X

a) n/m bits

b) m bits

• c) n bits

d) 1 bit

Un TLB suele tener:

Usuar Profesor

io es

X

a) tantas entradas como número de páginas la memoria física

• b) un número de entradas o elementos mucho menor que el número de páginas de la memoria virtual

c) más entradas que el número de páginas de la memoria virtual

d) una única entrada correspondiente a la última página accedida

Puntuación: -0,33

Si tenemos una memoria asociativa de n palabras con m bits/palabra, ¿cuál es el tamaño del registro indicador?

Usuar Profesor

io es

• a) n bits

b) m bits

X c) n/m
bits

d) m/n
bits

Puntuación: **-0,33**

Las señales BHE# y A0(=BLE#) son necesarias para seleccionar direcciones pares e impares en el procesador

Usuar Profesor

io es

a) 8088

✓ • b) 8086
c) 80386
d) Pentium
m

Puntuación: **1,00**

Supongamos una memoria asociativa con un registro indicador de x bits y otro de máscara de y bits. ¿Cuántas palabras puede almacenar dicha memoria?

Usuar Profesor

io es

- a) x
- b) y
- X** c) $x + y$
- d) Ninguna de las anteriores respuesta es cierta

Puntuación: **-0,33**

En una jerarquía de memoria, a medida que nos alejamos de la CPU:

Usuar Profesor

io es

- a) el tamaño de la unidad de transferencia entre dos niveles aumenta
- b) el tiempo de transferencia disminuye
- c) el tamaño de la memoria disminuye
- X** d) el coste por byte aumenta

Puntuación: **-0,33**

Aunque en general el entrelazado de memoria de orden inferior provoca menos conflictos en el acceso a memoria, ¿en cuál de los siguientes casos un entrelazado de orden superior provocaría menos conflictos?

$N = 2^n$ palabras, $M = 2^m$ módulos, módulo i, $0 \leq i \leq M$

Usuar Profesor

io es

- ✓ • a) Información distribuida en posiciones $i * 2^{n-m}$
b) En ningún caso
c) En cualquier caso
d) Información distribuida en posiciones consecutivas

Puntuación: **1,00**

El tamaño del registro de salida de una memoria asociativa de n palabras y m bits/palabra es:

Usuar Profesor

io es

- a) $n*m$ bits
b) $n+m$ bits
c) n bits
d) m bits

Puntuación: **1,00**

Se tiene una memoria que emplea entrelazado. Si fallan varias celdas contiguas de uno de sus chips de memoria, ¿con qué tipo de entrelazado de memoria sería más fácil poder utilizarla?

Usuar Profesor

io es

- ✓ • a) Entrelazado de orden superior
b) Entrelazado de orden inferior
c) Entrelazado de orden intermedio
d) Es igual de difícil en cada caso

Puntuación: **1,00**

Supongamos una memoria asociativa con un registro indicador de x bits y otro de máscara de y bits. ¿Cuántas palabras puede almacenar dicha memoria?

Usuar Profesor

io es

- ✓ • a) x

- b) y
- c) $x + y$
- d) Ninguna de las anteriores respuesta es cierta

Puntuación: **1,00**

La memoria virtual del computador es:

Usuar Profesor

io es

- a) Más rápida que la memoria principal.
- b) De menor capacidad que la memoria principal.
- c) a) y b) son correctas.
- ✓ • d) Ninguna de las anteriores es correcta.

Puntuación: **1,00**

¿Qué tipo de información no contiene una entrada de la tabla de páginas cuando utilizamos memoria virtual paginada?

Usuar Profesor

io es

- a) Tipo de acceso permitido
- b) Dirección del marco de página que ocupa
- ✗ c) Bit indicador de si la página es valida
- d) Longitud de la página

Puntuación: **-0,33**

En una caché con 64 bytes de longitud de línea, ¿cuál es la dirección de memoria de la primera palabra de la línea de caché que contenga la posición de memoria 0xBEE3DE72?

Usuar Profesor

io es

- a) 0xBEE3DE6
E
- ✗ b) 0xBEE3DE7
0
- c) 0xBEE3DE4
0
- d) 0x0EE3DE7
2

Puntuación: **-0,33**

[T6.2]

¿Cuántas líneas de dirección son necesarias en un memoria RAM dinámica de 256 K palabras? ¿Y en una estática?

Usuario Profesores

✓ • a)

9/18

b)

9/9

c)

18/9

d)

18/18

Puntuación: 1,00

2

Elección única

[T6.1]

Un sistema de memoria tiene un tiempo medio de acceso de 10 ns por operación de lectura o escritura y un ancho de datos de 32 bits. ¿Cuál es el ancho de banda del sistema de memoria?

Usuario Profesores

a)

2,5 ns

b)

32 Mbytes por segundo

✓ • c)

400 millones de bytes por segundo

d)

32 bits

Puntuación: 1,00

3

Elección única

[T4.4]

Los riesgos de datos consisten en que...

Usuario Profesores

a)

dos instrucciones acceden a la vez al mismo dato

✓ • b)

una instrucción necesita un dato calculado por otra anterior

c)

dos instrucciones necesitan leer el mismo dato

d)

todas las respuestas anteriores son correctas

Puntuación: 1,00

4

Elección única

Los procesadores comerciales con unidad de control microprogramada suelen almacenar los microprogramas...

Usuario Profesores

- a) en una PLA.
- b) en una RAM.
- c) en un banco de registros.
- d) en una ROM.

Puntuación: 1,00

5

Elección única

En una jerarquía de memoria, a medida que nos alejamos del procesador...

Usuario Profesores

- a) El coste por byte aumenta.
- b) La velocidad de transferencia aumenta.
- c) El tamaño de la memoria disminuye.
- d) El tamaño de la unidad de transferencia entre dos niveles aumenta.

Puntuación: 1,00

6

Elección única

[T6.1]

¿Cuál de las siguientes afirmaciones acerca de una jerarquía de memoria es cierta?

Usuario Profesores

- a) Para aumentar la eficiencia se transfieren bloques completos
- b) Toda la información que la CPU necesita está en el nivel 1
- c) Si una palabra no se encuentra en el tercer nivel entonces se busca en el segundo nivel
- d) Todas las afirmaciones anteriores son falsas

Puntuación: 1,00

7

Elección única

Aunque en general el entrelazado de memoria de orden inferior provoca menos conflictos en el acceso a memoria, ¿en cuál de los siguientes casos un entrelazado de orden superior provocaría menos conflictos?

$N = 2^n$ palabras, $M = 2^m$ módulos, módulo i , $0 \leq i \leq M$

Usuario Profesores

a)
En cualquier caso
 • b)
Información distribuida en posiciones $i * 2^{n-m}$
c)
En ningún caso
d)
Información distribuida en posiciones consecutivas
Puntuación: 1,00

8

Elección única

[T6.1]
¿Qué dice la ley de Moore?

Usuario Profesores

a)
Que el tamaño de los transistores se duplica cada 18 meses.
b)
Que la memoria de los ordenadores se duplica cada 18 meses.
 • c)
Que las prestaciones de los transistores en un chip se duplican cada 18 meses.
d)
Todas las respuestas son ciertas.
Puntuación: 1,00

9

Elección única

Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro

Usuario Profesores

 • a)
De instrucción
b)
Contador de programa
c)
De direcciones
d)
Puntero de pila
Puntuación: 1,00

10

Elección única

En la captación de la instrucción:

Usuario Profesores

a)
en MBR indicamos la dirección donde está la instrucción y en IR recogemos la instrucción
b)

en MBR indicamos la dirección donde está la instrucción y en MAR recogemos la instrucción

c)

en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción

✓ • d)

en MAR indicamos la dirección donde está la instrucción y en MBR recogemos la instrucción

Puntuación: 1,00

Sobre la segmentación:

Usuario Profesores

✓ • a)

Existen limitaciones al rendimiento provocadas por las instrucciones de salto y por las dependencias de datos.

b)

Un procesador superescalar no puede estar segmentado.

c)

La frecuencia de reloj viene impuesta por la etapa más corta.

d)

Es una técnica para comenzar simultáneamente la ejecución de varias instrucciones con el fin de reducir el tiempo de ejecución.

Puntuación: 1,00

2

Elección única

Una memoria que está estructurada en palabras de 8 bits tiene una capacidad de 32 Kbits.

¿Cuántas líneas de dirección tiene dicha memoria?

Usuario Profesores

a)

4

✓ • b)

12

c)

8

d)

32

Puntuación: 1,00

3

Elección única

¿En qué tipo de traducción de memoria virtual se utilizan los campos base y límite?

Usuario Profesores

✓ • a)

Segmentación

b)

Paginación

c)

Tanto en a) como en b)

d)

Ni en a) ni en b)

Puntuación: 1,00

4

Elección única

Suponga que la micropalabra de una máquina microprogramada tiene 8 bits de ancho y se usan 16 micropalabras diferentes en un microprograma de 256 micropalabras. Si se usa nanoprogramación...

Usuario Profesores

a)

no se ahorran bits pero el funcionamiento es más rápido.

b)

no se ahorran bits y además el funcionamiento es más lento.

✓ • c)

se ahorran bits pero el funcionamiento es más lento.

d)

se ahorran bits y el funcionamiento es más rápido.

Puntuación: 1,00

5

Elección única

[T4.3]

Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para escribir el resultado en el banco de registros, ¿cuál es la frecuencia de reloj máxima del procesador?

Usuario Profesores

a)

40 MHz

X b)

500 MHz

• c)

66,67 MHz

d)

200 MHz

Puntuación: -0,33

6

Elección única

[T6.2]

Una memoria que está organizada en palabras de 8 bits tiene una capacidad de 32 Kbits.

¿Cuántas líneas de dirección tiene dicha memoria?

Usuario Profesores

a)

4

b)

32

c)

8

✓ • d)

12

Puntuación: 1,00

7

Elección única

Los procesadores comerciales con unidad de control microprogramada suelen almacenar los micropogramas...

Usuario Profesores

✓ • a)

en una ROM.

b)

en una RAM.

c)

en un banco de registros.

d)

en una PLA.

Puntuación: 1,00

8

Elección única

[T6.5]

En las políticas anticipativas de extracción de cache, ¿cuál de ellas se caracteriza por preextraer el bloque $i+1$ si se referencia al bloque i y se produce falta de bloque?

Usuario Profesores

a)

Preextracción indexada

b)

Preextracción marcada

✓ • c)

Preextracción por falta

d)

Preextracción siempre

Puntuación: 1,00

9

Elección única

En el Pentium, el TLB permite buscar rápidamente...

Usuario Profesores

• a)

La dirección física a partir de la dirección lineal

b)

El descriptor de un segmento a partir de un selector

X c)

La dirección lineal a partir de la dirección virtual

d)

El contenido de un registro caché a partir de un selector

Puntuación: -0,33

10

Elección única

Un sistema no segmentado tarda 200 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce segmentado de 20 etapas con un ciclo de reloj de 12 ns. Cuando se

procesan muchas tareas, la máxima ganancia de velocidad que podría obtenerse se acerca a:

Usuario Profesores

a)

1,667

b)

1,2

• c)

16,67

x d)

20

Puntuación: -0,33

Si queremos almacenar la palabra de 16 bits 0x8965 en memoria según "big-endian", quedará almacenada a partir de la posición 0x1000 como:

Usuar Profesor

io es

- ✓ • a) en el byte 0x1000 se guarda 0x89 y en el 0x1001 0x65
b) en el byte 0x1000 se guarda 0x91 y en el 0x1001 0xA6
c) en el byte 0x1000 se guarda 0xA6 y en el 0x1001 0x91
d) en el byte 0x1000 se guarda 0x65 y en el 0x1001 0x89

Puntuación: **1,00**

Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?

Usuar Profesor

io es

- ✓ • a) $k + n \cdot 2^m$
b) $k + n^m$
c) $k + n \cdot (2^m - 1)$
d) Ninguno de los anteriores

Puntuación: **1,00**

Un procesador con una unidad de control microprogramada tiene una memoria de control de 256 palabras de 16 bits, de las que 128 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?

Usuar Profesor

io es

- a) 4096 bits
 b) No se produce ahorro
• c) 256 bits
d) 3840 bits

Puntuación: **-0,33**

¿Cómo actúa el indicador N del registro de indicadores de estado?

Usuar Profesor

io es

- a) Se pone a 1 cuando el resultado es negativo.
b) Se pone a 0 cuando el resultado es negativo.
c) Se pone a 1 cuando el resultado es positivo.
d) Se pone a 1 cuando el resultado de una operación es 0.

Puntuación: **1,00**

¿En qué tipo de ciclo de refresco RAS# permanece a 1?

Usuar Profesor

io es

- a) Sólo RAS#
b) CAS# antes de RAS#
 c) Refresco transparente
• d) En ninguno de los anteriores

Puntuación: **-0,33**

Respecto a las unidades de control nanoprogramadas:

Usuar Profesor

io es

- a) El diseño de las unidades de control nanoprogramadas debe ser vertical.
- b) Suponiendo una memoria de microprograma con n microinstrucciones de w bits cada una, de las cuales 2^m son distintas, el ahorro en bits si se utiliza nanoprogramación es $(n \cdot m + 2^m \cdot w) - n \cdot w$.
- X c) La realización nanoprogramada de una unidad de control es más rápida que la microprogramada.
- d) La anchura de la memoria de nanoprograma es la misma que la de memoria de microprograma en un diseño de la misma unidad de control que no usara nanoprogramación.

Puntuación: **-0,33**

¿En qué tipo de ciclo de refresco se hace RAS# = 0?

Usuar Profesor

io es

- a) Sólo RAS#
- b) CAS# antes de RAS#
- X c) Refresco transparente
- d) En todos los anteriores

Puntuación: **-0,33**

El ancho de banda de memoria es:

Usuar Profesor

io es

- a) el número de bits que se pueden transferir entre ésta y la CPU en paralelo en una sola operación de lectura o escritura
- b) el tiempo que se tarda en transferir una palabra entre memoria y CPU
- c) el intervalo de frecuencias de reloj permitidas entre memoria y CPU
- ✓ • d) ninguna de las anteriores es cierta

Puntuación: **1,00**

¿En qué tipo de refresco de memoria DRAM CAS# permanece a 0 después del ciclo de lectura o escritura precedente?

Usuar Profesor

io es

- a) RAS# antes de CAS#
- b) Sólo RAS#
- c) Refresco transparente
- d) Ninguna de las anteriores respuestas es correcta

Puntuación: **1,00**

¿En qué registro está contenido el último dato (o instrucción) leído de memoria, o el dato que se va a escribir en memoria?

Usuar Profesor

io es

- a) Acumulador
- b) PC.
- c) MBR.
- d) MAR.

Puntuación: **1,00**

En una arquitectura RISC típica:

Usuar Profesor

io es

- a) se usa un porcentaje elevado de las instrucciones del repertorio.
- b) la programación resulta mucho más simple que en una arquitectura CISC.
- c) no puede usarse segmentación.
- d) la UC es más compleja que en una arquitectura CISC.

Puntuación: **-0,33**

¿Cuál de las siguientes afirmaciones es falsa?

Usuar Profesor

io es

- a) La operación de lectura de una celda DRAM es destructiva
- b) Las memorias DRAM son en general mucho más rápidas que las SRAM
- c) Una celda DRAM sólo necesita un transistor y un

condensador

- d) Las memorias DRAM presentan generalmente una capacidad de almacenamiento mucho mayor que las SRAM

Puntuación: **1,00**

¿Cuál es el tamaño de la marca de caché en un ordenador capaz de direccionar 1 MB de memoria principal y 32 KB de memoria caché, que emplea un tamaño de palabra de 32 bits y correspondencia asociativa por conjuntos con 16 palabras por bloque y 8 bloques por conjunto, suponiendo que las direcciones de memoria utilizan 20 bits?

Usuar Profesor

io es

- a) 10 bits
- b) 7 bits
- ✓ • c) 8 bits
- d) 6 bits

Puntuación: **1,00**

¿En qué tipo de refresco de memoria DRAM CAS# permanece a 0 después del ciclo de lectura o escritura precedente?

Usuar Profesor

io es

- a) RAS# antes de CAS#
- b) Sólo RAS#
- ✓ • c) Refresco transparente
- d) Ninguna de las anteriores respuestas es correcta

Puntuación: **1,00**

Nombre:
DNI:
Grupo:

Examen Test (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. Si queremos almacenar la palabra de 64bits 0x00000001f ffffffe0 en una memoria de bytes según la convención little-endian a partir de la posición 0x0804913c, quedará
 - a. 0x00 en 0x0804913c y 0xe0 en 0x08049143
 - b. 0x1f en 0x0804913c y 0xe0 en 0x08049140
 - c. **0xe0 en 0x0804913c y 0x1f en 0x08049140**
 - d. Todas las respuestas anteriores son incorrectas

2. La diferencia entre el flag de acarreo y de overflow es que...
 - a. uno se activa cuando se opera con números con signo y otro cuando son sin signo
 - b. el flag de acarreo indica que ha habido acarreo en una operación con números enteros (ints), el de overflow indica que ha habido desbordamiento en una operación con números en punto flotante (p.f.)
 - c. **ambos se recalculan tras cada operación aritmético-lógica con ints, correspondiendo al programador consultar uno u otro según piense que sus datos son con o sin signo**
 - d. el de acarreo indica que el resultado es demasiado grande (para p.f.) o positivo (si se trata de ints) para poder almacenarse, el de overflow indica que es demasiado pequeño (p.f.) o negativo (ints)

3. ¿Cuál de las siguientes secuencias de instrucciones multiplica el (contenido del) registro EAX por 18?
 - a. imull \$0x18, %eax
 - b. **leal (%eax,%eax,8), %eax**
leal (%eax,%eax), %eax
 - c. shll \$18, %eax
 - d. sarl \$1, %eax
imul \$9, %eax

4. La instrucción IA32 test sirve para...
 - a. Testear el código de condición indicado, y poner un byte a 1 si se cumple
 - b. Mover el operando fuente al destino, pero sólo si se cumple la condición indicada
 - c. Realizar la operación resta (a-b) pero no guardar el resultado, sino simplemente ajustar los flags
 - d. **Realizar la operación and lógico bit-a-bit (a&b) pero no guardar el resultado, sino simplemente ajustar los flags**

5. La instrucción leave equivale a:
 - a. movl %esp, %ebp; popl %esp
 - b. **movl %ebp, %esp; popl %ebp**
 - c. pushl %esp; movl %ebp, %esp
 - d. pushl %ebp; movl %esp, %ebp

6. Alguna de estas opciones contiene algún elemento que no corresponde a los contenidos de un marco de pila GCC/Linux IA32.
- Argumentos de llamada a la función y Dirección de retorno
 - Antiguo marco de pila y Registros salva-invocante
 - Variables locales y Registros salva-invocado
 - Variables globales y Valor de retorno de la función**
-
7. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?
- EAX
 - EBX**
 - ECX
 - EDX
-
8. La arquitectura x86-64 tiene:
- 8 registros de propósito general (RPG) de 64 bits (%rax, %rbx, ... %rsp, %rbp)
 - 16 registros RPG de 64 bits**
 - 32 registros RPG de 64 bits
 - 64 registros RPG de 64 bits
-
9. ¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso? (salva-invocante, invocado, etc)
- RAX, RBX, RCX, RDX
 - RBX, RSI, RDI
 - RSP, RBP
 - CL, DX, R8d, R9**
-
10. La zona roja en x86-64 Linux es...
- una zona de pila en donde no deben escribir las funciones invocadas (porque si se produjera una interrupción, el manejador de interrupción sobreescribiría los valores escritos en pila)
 - una zona de pila en donde pueden escribir las funciones invocadas, pero teniendo en cuenta que los valores escritos pueden verse alterados si se produce una interrupción
-
- c. **una zona bajo (RSP) (adonde apunta RSP) que una función puede usar sin reservarla, pero sólo si no llama a ninguna otra función mientras la usa**
- d. una zona bajo (RBP) (adonde apunta RBP) que una función puede usar sin reservarla, pero sólo si no llama a ninguna otra función mientras la usa
-
11. Alguna de las siguientes *no* es una operación básica de la unidad de control:
- Transferir un registro a otro
 - (Leer o escribir) un registro (de / a) memoria
 - (Guardar o recuperar) un registro (en / de) la pila**
 - Realizar operación ALU y guardar resultado en registro
-
12. Una cola de precaptación sirve para:
- Reducir el efecto de los fallos de cache**
 - Disminuir el periodo de ciclo del cauce
 - Aumentar el número de etapas del cauce
 - Resolver ciertos problemas de dependencia de datos
-
13. Sólo una de las siguientes afirmaciones sobre memorias ROM es correcta. ¿Cuál?
- Para fabricar una ROM se deben conocer los datos que se desea que almacene**
 - Una PROM (Programmable ROM) se puede grabar usando un dispositivo programador que selectivamente funde contactos aplicándoles altas temperaturas mediante diminutas cabezas soldadoras (“equipo de puntas”)
 - Una EEPROM (Electrically Progr. ROM) se puede grabar eléctricamente, sin fundir contactos, pero no se puede borrar
 - Una EEPROM (Erasable EEPROM) se puede grabar (eléctricamente), y borrar (usando rayos ultravioleta)

ENCENDER TU LLAMA CUESTA MUY POCO



14. Una SRAM de 32Kx8bit (256Kbit) puede venir organizada en 512 filas, dedicando por tanto al decodificador de columnas...

- a. 6 bits
- b. 7 bits
- c. 8 bits
- d. 9 bits

15. Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta

- a. 64 pastillas
- b. 32 pastillas
- c. 16 pastillas
- d. 8 pastillas

16. Una sentencia en C del tipo “`while (test)
body;`” puede transformarse en código “`goto`” como:

- a.

```
if (!test) goto done;  
loop: body;  
      if (test) goto loop;  
done:
```
- b.

```
loop: body;  
      if (test) goto loop;
```
- c.

```
if (test) goto true;  
      goto done;  
true: body;  
done:
```
- d.

```
loop: if (test) goto done;  
      body;  
      goto loop;  
done:
```

17. El cuerpo del siguiente código C:

```
unsigned copy(unsigned u) {return u;}
```

puede traducirse a ensamblador como:

- a. `movl 8(%ebp), %eax`
- b. `movl %ebp, 8(%eax)`
- c. `movl 8(%ebp), (%eax)`
- d. `movl 8(%esp), %ebp`

18. Si representamos la fase Decode con una D, Execute con una E, Fetch con una F y Writeback con una W, el orden correcto de

las distintas fases de una instrucción máquina es:

- a. D E F W
- b. F D E W
- c. F W D E
- d. D F E W

19. Si N es el número de instrucciones máquina de un programa, F es la frecuencia de reloj, y C el número promedio de ciclos por instrucción, el tiempo de ejecución del programa será:

- a. $N \cdot F/C$
- b. $N \cdot F \cdot C$
- c. `N·C/F`
- d. $N/(F \cdot C)$

20. ¿Cuál de las siguientes expresiones representa un direccionamiento inmediato?

- a. `%eax`
- b. `$0x400`
- c. `(%eax)`
- d. `8(%ebp)`

21. La instrucción `movzbl %al, %eax`

- a. Pone a 0 el registro `%eax`
- b. Copia en `%eax` el valor sin signo almacenado en `%al`, llenando con ceros
- c. Copia en `%eax` el valor del indicador de cero
- d. Copia en `%eax` el valor de `%al` si el indicador de cero está activado

22. Dentro de una función declarada como `void swap(int *xp, int *yp)`, que intercambia los valores de los dos enteros cuyas direcciones de memoria (punteros) son pasadas como parámetros a la función, la instrucción `movl 12(%ebp), %ecx` copia en `%ecx`...

- a. el valor del entero apuntado por el puntero pasado como primer parámetro
- b. el valor del entero apuntado por el puntero pasado como segundo parámetro

- c. el valor del puntero pasado como primer parámetro
- d. el valor del puntero pasado como segundo parámetro
-
23. Si la dirección del primer elemento de un vector de enteros z está almacenada en el registro %edx y la variable entera i está almacenada en el registro %eax, la instrucción máquina que realiza la operación $z[i]++$ es:
- a. addl \$1, (%edx,%eax,4)
- b. addl \$1, (%eax,%edx,4)
- c. addl \$1, (%edx,%eax)
- d. addl \$4, (%eax,%edx)
-
24. En un camino de datos con un solo bus, para realizar la operación de copia de un registro r1 en un registro r2, es decir $r2 \leftarrow r1$, es necesario:
- a. Activar la carga del registro r1 y habilitar la salida triestado del registro r2
- b. Habilitar la salida triestado del registro r1 y activar la carga del registro r2
- c. Habilitar las salidas triestado de los registros r1 y r2 y activar la carga del registro r2
- d. Habilitar la salida triestado del registro r2 y activar la carga de los registros r1 y r2
-
25. La salida de un campo del registro de microinstrucción que solapa dirección de salto y algunas señales de control han de conectarse a:
- a. una ROM o PLA
- b. la memoria de control
- c. el registro de instrucción
- d. un demultiplexor controlado por el tipo de salto
-
26. Cuando dos o más instrucciones necesitan un recurso hardware en el mismo ciclo, se trata de un riesgo:
- a. estructural
- b. por dependencia de datos
- c. de control
-
- d. de salto
-
27. Una instrucción típica de entrada / salida tiene
- a. no tiene ningún parámetro
- b. tiene un parámetro: un registro del procesador
- c. tiene dos parámetros: un registro del procesador y una dirección de puerto de E/S
- d. tiene tres parámetros: un registro del procesador, una dirección de puerto de E/S y una dirección de memoria
-
28. ¿Cuál de las siguientes afirmaciones acerca del daisy-chain es cierta?
- a. Todos los componentes se conectan directamente y con igual prioridad al procesador o gestor de interrupciones
- b. Los componentes se comportan de forma cooperativa: sólo al de mayor prioridad se le concede la interrupción o se apodera del bus de comunicaciones
- c. Los componentes están conectados todos con todos y un gestor centralizado decide la prioridad
- d. Es incompatible con la técnica de sondeo o polling
-
29. La memoria DRAM:
- a. Se denomina dinámica porque para mantener almacenado un dato hay que recargarlo cada cierto tiempo en un ciclo de refresco
- b. Es menos densa que la memoria SRAM
- c. Se inventó en la década de los 90
- d. Necesita 6 transistores por cada celda
-
30. Los módulos de memoria dinámica compactos que suelen usarse en los portátiles se denominan:
- a. SIMM
- b. SODIMM
- c. SLIM
- d. MIN
-

Nombre:

DNI:

Grupo:

Examen de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. ¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa en lenguaje C? (el fichero sin extensión es un ejecutable):
 - a. fich.c -> fich.o -> fich.s -> fich
 - b. fich -> fich.s -> fich.o -> fich.c
 - c. fich.c -> fich.s -> fich -> fich.o
 - d. fich.c -> fich.s -> fich.o -> fich**

 2. ¿Qué hace gcc -O?
 - a. **Compilar con optimización suave**
 - b. Compilar .c->.o (fuente C a objeto)
 - c. Compilar .s->.o (fuente ASM a objeto)
 - d. Ambas (b) y (c), según la extensión de los ficheros que se usen como argumentos

 3. ¿Qué modificador (switch) de gcc hace falta para compilar .c->.s (de fuente C a fuente ASM)?
 - a. Eso no se puede hacer con gcc
 - b. gcc -c
 - c. gcc -s
 - d. gcc -S**

 4. ¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?
 - a. -m32
- b. -32
 - c. -m elf_i386**
 - d. No hace falta modificador, ld lo deduce del tipo de objeto a enlazar
-
5. Compilar .c->exe (de fuente C a ejecutable) usando sólo as y ld, sin gcc...
 - a. **No se puede**
 - b. Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
 - c. Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C
 - d. Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C

 6. ¿Cuál de las siguientes afirmaciones es falsa respecto al lenguaje C?
 - a. En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina
 - b. Los parámetros se introducen en la pila en el orden inverso a como aparecen en la llamada de C, es decir, empezando por el último y acabando por el primero
 - c. Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la**

pila los parámetros de llamada realizando varios pop

- d. Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la dirección de memoria donde está almacenada la variable

7. El primer parámetro de printf:

- a. es un entero
b. es un número en coma flotante
c. es un puntero
d. puede ser de cualquier tipo

8. Para averiguar la paridad de un número se puede usar la operación:

- a. NOT
b. AND
c. OR
d. XOR

9. Utilizando la sentencia asm, las denominadas restricciones que se indican al final de dicha sentencia, involucran a:

- a. Solamente las entradas
b. Solamente las salidas
c. Solamente los sobrescritos
d. Ninguna de las anteriores es cierta

10. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%r"
    :[r] "+r" (result)
    :[i] "r" (i),
    [a] "r" (array));
```

¿Cuál de las siguientes afirmaciones es correcta?

- a. r es una posición de memoria de entrada/salida
b. a es una posición de memoria de entrada
c. i es un registro de entrada
d. la salida de la función se fuerza a que esté en la variable result

11. En una bomba como las estudiadas en la práctica 4, del tipo...

```
0x0804873f <main+207>: call 0x8048504 <scanf>
0x08048744 <main+212>: mov 0x24(%esp),%edx
0x08048748 <main+216>: mov 0x804a044,%eax
0x0804874d <main+221>: cmp %eax,%edx
0x0804874f <main+223>: je 0x8048756 <main+230>
0x08048751 <main+225>: call 0x8048604 <boom>
0x08048756 <main+230>: ...
```

la contraseña es...

- a. el entero 0x804a044
b. el entero almacenado a partir de la posición de memoria 0x804a044
c. el string almacenado a partir de la posición de memoria 0x24(%esp)
d. ninguna de las anteriores

12. En una bomba como las estudiadas en la práctica 4, del tipo...

```
0x08048705 <main+149>: call 0x80484c4 <gettimeofday>
...
0x08048718 <main+168>: cmp $0x5,%eax
0x0804871b <main+171>: jle 0x8048722 <main+178>
0x0804871d <main+173>: call 0x8048604 <boom>
0x08048722 <main+178>: ...
```

ejecutada paso a paso con el depurador ddd, interesaría...

- a. ejecutar hasta jle, ajustar %eax a 6, y continuar ejecutando paso a paso
b. ejecutar hasta jle, ajustar %eax a 4, y continuar ejecutando paso a paso
c. cambiar jle por jmp usando ddd o un editor hex, salvar el programa y reiniciar la depuración con el nuevo ejecutable
d. Ninguna de las opciones anteriores es de interés (bien porque no se pueda hacer eso o porque no sirva para evitar la bomba)

13. En una bomba como las estudiadas en la práctica 4, del tipo...

Estudialo bien, que
tiene pinta de importante

Esto lo pregunta
seguuuuuuuro!

Echa un vistazo
verás que guay



Clases en
DIRECTO



■ Audio y video
PROFESIONAL



Pizarra digital
COMPARTIDA



**Máximo
10 PERSONAS**

 @AcademiaOpensky

Una página más, y a por un café

Ánimo. Tu puedes

Ánimo, tu puedes

```
0x080486e8 <main+120>: call 0x8048524 <strcmp>
0x080486ed <main+125>: test %eax,%eax
0x080486ef <main+127>: je 0x80486f6 <main+134>
0x080486f1 <main+129>: call 0x8048604 <boom>
0x080486f6 <main+134>: ...
```

la contraseña es...

- a. el valor que tenga %eax
 - b. el string almacenado a partir de donde apunta %eax
 - c. el entero almacenado a partir de donde apunta %eax
 - d. ninguna de las anteriores**

14. Respecto a las bombas estudiadas en la práctica 4, ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la contraseña? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" podemos entender la cifra del César, por ejemplo. "Invertir" es darle la vuelta al string de manera que la primera letra se convierta en la última y viceversa.

- a. 1 string del usuario se cifra, y se compara con el string del fuente
 - b. 2 strings del fuente se invierten, se concatenan, se cifra el resultado, y se compara con el string del usuario
 - c. 2 strings del fuente se concatenan, se invierte el resultado, se cifra, y se compara con el string del usuario
 - d. Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

15. Respecto a las bombas estudiadas en la práctica 4, ¿en cuál de los siguientes tipos de bomba sería más difícil descubrir la(s) contraseña(s)? Se distingue entre enteros definidos en el código fuente de la bomba, y enteros solicitados al usuario mediante `scanf()`. Por "procesar" podemos entender

calcular el n-ésimo elemento de la serie de Fibonacci, por ejemplo.

- a. 1 entero del fuente se procesa, y se compara con el entero del usuario
 - b. 2 enteros del usuario se suman, se procesa la suma, y se compara con el entero del fuente
 - c. 2 enteros del usuario se procesan, se suman los resultados, y se compara con el entero del fuente
 - d. Las 2 (o 3) opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

16. Respecto al procesador que denominamos ssrDLX (procesador DLX sin segmentar), ¿cuál de las siguientes afirmaciones es correcta?

- a. Todas las instrucciones deben realizar las cuatro etapas del cauce
 - b. Las etapas ID y MEM duran un 80% del tiempo de ciclo de reloj
 - c. Una operación de suma de enteros (operando y resultado en el banco de registros) necesitará 3.8 ciclos en el procesador sin segmentar

- d. **ld f2,a** en el procesador ssrDLX requiere **4.6 ciclos**

17. Dado un programa en un procesador DLX, se puede aplicar la técnica que recibe el nombre de desenrollado de bucle, la cual repercute en:

- a. realizar los cálculos de varias iteraciones en diferentes subrutinas
 - b. reducir el número de instrucciones de salto que se tienen que ejecutar**
 - c. aumentar el número de instrucciones en el código para aumentar la probabilidad de que existan riesgos de control
 - d. reorganización de las instrucciones para reducir el efecto de las dependencias entre ellas

¿Cuál de las anteriores afirmaciones es cierta?

18. La directiva .text en el simulador DLX con la dirección 256, (.text 256), indica en un programa que:

- a. la variable text tiene el valor 256
 - b. La primera instrucción del programa se ubicará en la posición 0x100**
 - c. Existe una etiqueta de salto denominada .text en la posición 256
 - d. Todas las anteriores afirmación son incorrectas
-

19. ¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?

- a. posición 0: FF pos.1:FF pos.2: FF pos.3: 00
 - b. 0:00 1:FF 2:FF 3:FF
 - c. 0:00 1:01 2: 00 3:80
 - d. Ninguna de las anteriores**
-

20. En un procesador de la familia 80x86 las posiciones de memoria que representan una variable entera contiene los bytes: F0 FF FF FF. ¿Cuánto vale dicha variable?

- a. -16**
 - b. 4043309055
 - c. 16
 - d. 4294967280
-

Nombre:
DNI:
Grupo:
Examen Test (3.0p)
Tipo A

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. ¿Cuál de las siguientes direcciones está alineada a double (8-byte)?

- a. 1110110101110111_2
- b. 1110110101110100_2
- c. 1110110101110000_2
- d. Ninguna de ellas

2. ¿Cómo se devuelve en ensamblador x86 Linux gcc el valor de retorno de una función al terminar ésta?

- a. La instrucción RET lo almacena en un registro especial de retorno
- b. Por convención se guarda en %eax
- c. Se almacena en pila justo encima del (%ebp) del invocado
- d. Se almacena en pila justo encima de los argumentos de la función

3. ¿Cuál afirmación es FALSA en arquitecturas x86-64?

- a. El tamaño de un double es 64 bits
- b. El tamaño de los registros es 64 bits
- c. El tamaño de un puntero es 64 bits
- d. El tamaño de las posiciones de memoria es 64 bits

4. Considerar las siguientes declaraciones de estructuras en una máquina Linux de 64-bit.

```
struct RECORD {           struct NODE {  
    long value2;           double value;  
    int ref_count;         struct RECORD record;  
    char tag[4];           char string[8];  
};                      };
```

También se declara una variable global my_node como sigue:
`struct NODE my_node;`

Si la dirección de my_node es 0x601040, ¿cuál es el valor de &my_node.record.tag[1] ?

- a. 0x601050

- b. 0x601054

- c. 0x601055

- d. Ninguna de las anteriores

5. En la pregunta anterior, ¿cuál es el tamaño de my_node en bytes?

- a. 32
- b. 40
- c. 28
- d. Ninguno de los anteriores

6. ¿Cuál de las siguientes instrucciones x86 se puede usar para sumar dos registros y guardar el resultado sin sobrescribir ninguno de los registros originales?

- a. mov
- b. lea
- c. add
- d. Ninguna de ellas

7. Respecto a los registros enteros en arquitectura IA-32

- a. Hay 8, y en cada uno puede accederse a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX) ó a los 8 LSBs (p.ej. AL)
- b. Hay 8 de cada tamaño (32, 16, 8 bits), aunque no todos los registros tienen versión en 8 y 16 bits
- c. No hay distintos tamaños, son sólo registros de 32 bits, como corresponde a dicha arquitectura
- d. Son de 32bits en las aplicaciones de 32bit, y de 64bits en las aplicaciones de 64bit

8. Respecto a direccionamiento a memoria en ensamblador IA-32 (sintaxis AT&T), de la forma D(Rb, Ri, S), sólo una de las siguientes afirmaciones es FALSA. ¿Cuál?

- a. El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)

ENCENDER TU LLAMA CUESTA MUY POCO



- b. Los registros base e índice (Rb y Ri) pueden ser cualesquiera de los 8 registros enteros (EAX...ESP)
- c. El factor de escala S puede ser 1, 2, 4, 8
- d. El desplazamiento D también puede ser el nombre de una variable (que se traduce por su dirección, de 4bytes)
-
9. Las siguientes afirmaciones sugieren que el tamaño de varios tipos de datos en C (usando el compilador gcc) son iguales tanto en IA-32 como en x86-64. Sólo una de ellas es FALSA. ¿Cuál?
- a. El tamaño de un `int` es 4 bytes
- b. El tamaño de un `unsigned` es 4 bytes
- c. **El tamaño de un `long` es 4 bytes**
- d. El tamaño de un `short` es 2 bytes
-
10. Estudiando el listado de una función C presuntamente compilada con gcc en modo 64bit (x86-64), nos dicen que la primera instrucción, `movl %eax, (%rdi)`, carga en EAX el valor adonde apunta el primer argumento.
- a. Está mal, porque EAX no se puede usar en modo 64bit, debería ser RAX
- b. **Está mal, porque EAX no se carga con ningún valor**
- c. Está mal, porque el primer argumento de una función C no se pasa en RDI
- d. Está bien, y pone a cero los 32 bits más significativos de RAX
-
11. ¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?
- a. La frecuencia de reloj del procesador.
- b. El número medio de ciclos de reloj por instrucción.
- c. La arquitectura del procesador.
- d. **El resultado de la ejecución de un conjunto de programas de prueba.**
-
12. ¿Cuál es la característica tecnológica principal de la segunda generación de computadores?
- a. Los circuitos integrados
- b. **Los transistores**
- c. La gran integración de los circuitos (VLSI)
- d. Las válvulas
-
13. ¿Cuál de las siguientes no es una característica de los computadores RISC?
- a. La decodificación de las instrucciones debe ser simple: un computador RISC debería emplear un único formato de instrucción
- b. Para acelerar el computador RISC se emplean técnicas de pipelining.
- c. **Las funciones que realizan los computadores RISC deben ser lo más complejas y potentes que sea posible.**
- d. Un computador RISC no debe emplear microprogramación.
-
14. ¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?
- a. `JMP dir`
- b. **JNE dir**
- c. `CALL dir`
- d. `RET`
-
15. ¿Cuál de las siguientes parejas de nemotécnicos de ensamblador corresponden a la misma instrucción máquina?
- a. `CMP` (comparar), `SUB` (restar).
- b. `JC` (saltar si acarreo), `JL` (saltar si menor, para números con signo).
- c. **JZ (saltar si cero), JE (saltar si igual).**
- d. `SAR` (desplazamiento aritmético a la derecha) / `SHR` (desplazamiento lógico a la derecha).
-
16. ¿Cuál de los siguientes grupos de instrucciones sólo modifican los indicadores de estado sin almacenar el resultado de la operación?
- a. `AND`, `OR`, `XOR`
- b. `ADC`, `SBB`
- c. **CMP, TEST**
- d. `IMUL`, `IDIV`
-
17. ¿Cuál de las siguientes características es típica de la microporgramación horizontal?
- a. Muchos campos solapados.
- b. **Ninguna o escasa codificación.**
- c. Microinstrucciones cortas.
- d. Escasa capacidad para expresar paralelismo entre microoperaciones.
-
18. ¿En qué pareja de registros están el dato/instrucción que se leerá o escribirá en memoria, y la dirección de memoria?
- a. MAR y ACUMULADOR
- b. IR y ACUMULADOR
- c. **MBR y MAR**
- d. MBR y PC
-
19. ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microporgrama correspondiente?
- a. **Una memoria.**
- b. Un contador.
- c. Un multiplexor.
- d. Un registro.
-
20. ¿Con cuál de los siguientes dispositivos tendría sentido utilizar E/S programada sin consulta de estado?
- a. **Salida a un display de 7 segmentos**
- b. Entrada desde un disco duro
- c. Salida a una impresora
- d. Con ningún dispositivo tiene sentido
-
21. ¿Cuál de las siguientes funciones no corresponde a un módulo de E/S?
- a. Comunicación con el microprocesador
- b. Comunicación con el dispositivo
- c. **Almacenamiento de programas**
- d. Almacenamiento temporal de datos
-
22. ¿Cuál de los siguientes grupos de instrucciones podrá pertenecer a un procesador con E/S mapeada en memoria?
- a. `IN, LOAD, OUT`
- b. `IN, LOAD, MOV`
- c. **LOAD, MOV, STORE**
- d. Ninguno de los anteriores

BURN.COM

#StudyOnFire

BURN
ENERGY DRINK

23. ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

- a. 2
- b. 3
- c. 4
- d. 5

24. ¿Cuántos bits hacen falta como mínimo para implementar tres niveles de inhibición de interrupciones (general, nivel y máscara) en un sistema con cuatro niveles de interrupción?

- a. 4
- b. 5
- c. 6
- d. 7

25. ¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?

- a. En las transferencias síncronas
- b. En las transferencias asíncronas**
- c. En ambas
- d. En ninguna

26. ¿Qué técnica de E/S requiere menos atención por parte del procesador?

- a. E/S programada
- b. E/S mediante acceso directo a memoria**
- c. E/S mediante interrupciones
- d. Todas requieren la misma atención

27. ¿A qué tipo de localidad de memoria hace referencia la siguiente afirmación: "si se referencia un elemento, los elementos cercanos a él serán referenciados pronto"?

- a. Localidad espacial**
- b. Localidad secuencial
- c. Localidad temporal
- d. Ninguna de las respuestas anteriores es correcta

28. ¿A qué tipo de memoria caché corresponde la siguiente afirmación: "permite que cualquier dirección se pueda almacenar en cualquier marco de bloque de caché"?

- a. Con correspondencia directa
- b. Totalmente asociativa**
- c. Asociativa por conjuntos
- d. Ninguna de las anteriores

29. ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM dinámicas es cierta?

- a. Los datos permanecen en cada celda indefinidamente
- b. Las celdas de almacenamiento son complejas
- c. Las operaciones de lectura no son destructivas
- d. Las operaciones de escritura sirven como operaciones de refresco**

30. Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?

- a. 31,25 microsegundos**
- b. 61 nanosegundos
- c. 8192 milisegundos
- d. 7,8125 microsegundos

Nombre:
DNI:
Grupo:

Examen de Prácticas (4.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 4/20 si es correcta, 0 si está en blanco o claramente tachada, -4/60 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. Considerar los siguientes dos bloques de código almacenados en dos ficheros distintos:

```
/* main.c */          /* func.c */  

int i = 0;           int i = 1;  

int main() {         void func() {  

    func();           printf("%d", i);  

    return 0;          }  

}
```

¿Qué sucederá cuando se compile, enlace y ejecute este código?

- a. Error al compilar o enlazar, no se obtiene ejecutable
 - b. Escribe “0”
 - c. Escribe “1”
 - d. A veces escribe “0” y a veces “1”
-

2. ¿Qué hace gcc -S?

- a. Compilar .c → .s (fuente C a fuente ASM)
 - b. Compilar .s → .o (fuente ASM a objeto)
 - c. Compilar optimizando tamaño (size), no tiempo
 - d. Compilar borrando del ejecutable la tabla de símbolos (strip)
-

3. ¿Qué modificador (switch) de gcc hace falta para compilar .c → .o (de fuente C a código objeto)?

- a. Eso no se puede hacer con gcc

- b. gcc -c
 - c. gcc -o
 - d. gcc -O
-

4. ¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32bits (fuente C) en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?

- a. -m32
 - b. -32
 - c. -m elf_i386
 - d. No se puede conseguir ese efecto con modificadores de gcc, porque el gcc por defecto (/usr/bin/gcc) es el de 64bits, se debe usar el gcc de la instalación alternativa (de 32bits)
-

5. ¿Qué modificador (switch) de as hace falta para ensamblar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?

- a. -m32
 - b. -32
 - c. -m elf_i386
 - d. Ninguna de las anteriores respuestas es correcta
-

6. Como parte del proceso de compilación de una aplicación en lenguaje C, enlazar .o → .exe (de objeto proveniente de fuente C a ejecutable) usando sólo as y ld, sin gcc...
- Se puede, repartiendo entre as y ld los modificadores (switches) que corresponda
 - Se puede, repartiendo modificadores entre as y ld, y añadiendo al comando ld el runtime de C
 - Basta usar ld, con los modificadores de gcc que corresponda, y añadiéndole el runtime de C
 - Ninguna de las anteriores respuestas es correcta
-

7. La práctica "popcount" debía calcular la suma de bits de los elementos de un array. Un estudiante entrega la siguiente versión de popcount4:

```
int popcount4(unsigned* array, int len)
{
    int i, j, res = 0;
    for(i = 0; i < len; ++i) {
        unsigned x = array[i];
        int n = 0;
        do {
            n += x & 0x01010101L;
            x >>= 1;
        } while(x);
        for(j = 16; j == 1; j /= 2){
            n ^= (n >>= j);
        }
        res += n & 0xff;
    }
    return res;
}
```

Esta función popcount4:

- Produce el resultado correcto
 - Fallaría con array={0,1,2,3}
 - Fallaría con array={1,2,4,8}
 - No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
-

8. La práctica "paridad" debía calcular la suma de paridades impar (XOR de todos los bits) de los elementos de un array. Un estudiante entrega la siguiente versión de parity6:

```
int parity6(unsigned * array, int len)
{
    int i, result = 0;
    unsigned x;
    for (i=0; i<len; i++){
        x = array[i];
        asm( "mov %[x], %%edx\n\t"
             "shr $16, %%edx\n\t"
             "shr $8, %%edx\n\t"
             "xor %%edx,%%edx\n\t"
             "setp %%dl\n\t"
             "movzx %%dl, %[x]\n\t"
             : [x] "+r" (x)
             :
             : "edx"
             );
        result += x;
    }
    return result;
}
```

Esta función parity6:

- Produce el resultado correcto
 - Fallaría con array={0,1,2,3}
 - Fallaría con array={1,2,4,8}
 - No es correcta pero el error no se manifiesta en los ejemplos propuestos, o se manifiesta en ambos
-
- Respecto a las bombas estudiadas en la práctica 3, ¿en cuál de los siguientes tipos de bomba sería más **fácil** descubrir la(s) contraseña(s)? Se distingue entre strings definidos en el código fuente de la bomba, y strings solicitados al usuario mediante scanf(). Por "cifrar" se entiende aplicar la cifra del César (sumar o restar una constante fija a los códigos ASCII).
 - 1 string del fuente se cifra, y se compara con el string del usuario
 - 2 strings del usuario se concatenan, se cifra el resultado y se compara con el string del fuente
 - 2 strings del usuario se cifran, se concatenan los resultados, y se compara con el string del fuente
 - Las opciones más fáciles son de la misma dificultad, así que no se puede marcar ninguna como la más fácil

ENCENDER TU LLAMA CUESTA MUY POCO



10. Respecto a las bombas estudiadas en la práctica 3, ¿en cuál de los siguientes tipos de bomba sería más **difícil** descubrir la contraseña? Se distingue entre enteros definidos en el código fuente de la bomba, y enteros solicitados al usuario mediante scanf(). Por "procesar" se entiende calcular el n-ésimo elemento de la serie de Fibonacci.

- a. 1 entero del usuario se procesa, y se compara con el entero del fuente
- b. 2 enteros del fuente se suman, se procesa la suma, y se compara el resultado con el entero del usuario
- c. 2 enteros del fuente se procesan, se suman los resultados, y se compara la suma con el entero del usuario
- d. Las opciones más difíciles son de la misma dificultad, así que no se puede marcar ninguna como la más difícil

11. ¿Cuál de los siguientes es el orden correcto en el ciclo de compilación de un programa escrito en lenguaje C? (el fichero sin extensión es un ejecutable):

- a. file.s → file.c → file → file.o
- b. file.c → file.s → file.o → file
- c. file.c → file.s → file → file.o
- d. file.s → file.c → file.o → file

12. ¿Qué hace gcc -O?

- a. Compilar sin optimización, igual que -O0
- b. Compilar .c → .o
- c. Compilar .s → .o
- d. **Compilar con optimización, igual que -O1**

13. ¿Qué modificador (switch) de gcc hace falta para compilar .s → .o sin llamar al enlazador?

- a. Eso no se puede hacer con gcc
- b. **gcc -c**
- c. gcc -s
- d. gcc -S

14. ¿Qué modificador (switch) de gcc hace falta para compilar una aplicación de 32 bits en un sistema de 64 bits?

a. **-m32**

b. -m64

c. -32

d. -64

15. Compilar .s → ejecutable, usando sólo as y ld, sin gcc...

- a. No se puede
- b. **Se puede, usando en as y ld los modificadores (switches) que corresponda**
- c. Basta usar as, con los modificadores que corresponda
- d. Basta usar ld, con los modificadores que corresponda

16. ¿Cuál de las siguientes afirmaciones es cierta respecto al lenguaje C?

- a. **En lenguaje C, al llamar a una subrutina o función se introducen los parámetros en la pila y después se realiza una llamada a la subrutina**
- b. Los parámetros se introducen en la pila en el orden en el que aparecen en la llamada de C, es decir, empezando por el primero y acabando por el último
- c. Antes de volver de la rutina llamada, el programa en C se encarga de quitar de la pila los parámetros de llamada realizando varios pop
- d. Pasar a una función un puntero a una variable se traduce en introducir en la pila el valor de la variable

17. El primer parámetro de printf:

- a. es un char
- b. es un entero
- c. **es un puntero**
- d. puede ser de cualquier tipo, incluso no existir

18. Respecto al procesador que denominamos ssrDLX (procesador DLX sin segmentar), ¿cuál de las siguientes afirmaciones es falsa?

- a. **Todas las instrucciones deben realizar las cuatro etapas del cauce**

- b. Las etapas ID y WB duran un 80% del tiempo de ciclo de reloj
 - c. Una operación de suma de enteros (operando y resultado en el banco de registros) necesitará 3.6 ciclos en el procesador sin segmentar
 - d. `ld f2,a` en el procesador ssrDLX requiere 4.6 ciclos
-

19. La directiva `.text` en el simulador DLX con la dirección 256, (`.text 256`), indica en un programa que:

- a. la variable `text` tiene el valor 0x100
 - b. Existe una etiqueta de salto denominada `.text` en la posición 0x100
 - c. **La primera instrucción del programa se ubicará en la posición 0x100**
 - d. Todas las anteriores afirmaciones son incorrectas
-

20. En un procesador de la familia 80x86 una variable de 32 bits, entera con signo, almacenada a partir de la dirección n contiene: 0xFF en la dirección n , 0xFF en la dirección $n+1$, 0xFF en la dirección $n+2$ y 0xF0 en la dirección $n+3$. ¿Cuánto vale dicha variable?

- a. -16
 - b. **-251658241**
 - c. 16
 - d. 4294967280
-

Nombre:
DNI:
Grupo:

Examen Test (3.0p)

Todas las preguntas son de elección simple sobre 4 alternativas.

Cada respuesta vale 3/30 si es correcta, 0 si está en blanco o claramente tachada, -1/30 si es errónea.

Anotar las respuestas (a, b, c o d) en la siguiente tabla.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

1. En una CPU de 32bits con memoria de bytes, el problema es que...
 - a. No tiene sentido, un registro no cabría en memoria
 - b. No hay problema, cuando se salva un registro a memoria se escribe en la posición deseada
 - c. Hay que usar 4 instrucciones de lectura (o escritura) para leer (o escribir) un registro completo
 - d. **Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU**

2. En una memoria de bytes que contuviera a partir de la posición 0 los valores 1,0,0,0,0xFE,0xFF,0xFF,0xFF, se puede decir que...
 - a. Hay una palabra de 16bit big-endian con valor 1 en la posición 0
 - b. **Hay una palabra de 16bit little-endian con valor 254 en la posición 3**
 - c. Hay una palabra de 32bit little-endian con valor -1 en la posición 4
 - d. Todas las respuestas anteriores son incorrectas

3. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):
 - a. movw %dx, (%eax)
 - b. **pushb \$0xFF**
 - c. movswl (%eax), %edx
 - d. movzbl %dl, %eax

4. La instrucción xor \$3, %eax tiene como resultado:
 - a. Poner a 0 los últimos 3 bits del registro EAX
 - b. **Cambiar 0<->1 (complemento a 1 de) los últimos 2 bits del registro EAX**
 - c. Poner a 1 el último bit del registro EAX
 - d. Ninguno de los anteriores resultados

5. De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...
 - a. El bucle for
 - b. El bucle while
 - c. **El bucle do-while**
 - d. La selección switch-case

6. GCC/Linux IA32 resuelve el ajuste de marco de pila mediante las instrucciones:
 - a. movl %esp, %ebp; popl %esp

- b. `movl %ebp, %esp; popl %ebp`
c. `pushl %esp; movl %ebp, %esp`
d. `pushl %ebp; movl %esp, %ebp`
-
7. Respecto a registros salva-invocante y salva-invocado en GCC/Linux IA32, ¿cuál de éstos es de distinto tipo que el resto?
- a. EAX
b. EBX
c. ESI
d. EDI
-
8. La convención de llamada Linux/GCC x86-32 considera, respecto a convenios de uso de registros:
- a. 3 registros salva-invocante, 3 registros salva-invocado, y 2 especiales**
b. 8 registros salva-invocante, 6 registros salva-invocado, y 2 especiales
c. Algunos registros salva-invocante, otros salva-invocado, uno especial
d. Algunos registros para pasar argumentos, otros salva-invocante, otros salva-invocado, dos especiales
-
9. El marco de pila en x86-64 Linux...
- a. no existe, porque RBP no es registro especial en x86-64
b. sólo se crea para funciones que invocan anidadamente a otra función (procedimientos padre, no hojas)
c. se crea para funciones en las que GCC no puede evitar que RBP baje más, como por ejemplo: que haya demasiadas variables locales (y no quepan en registros), o que haya que salvar algún registro salva-invocado
d. **se crea para funciones en las que GCC no puede evitar que RSP baje más, como por ejemplo: que haya que calcular la dirección de una variable local, o pasar más de 6 argumentos a otra función**
-
10. En un sistema de 32bits, ¿cuál de las siguientes expresiones C es equivalente a la expresión $(x[2] + 4)[3]$? Asumir que x se ha declarado como `int **x`. Recordar que C usa aritmética de punteros. Notar que muchos de los paréntesis no son necesarios, sólo se han añadido para evitar confusiones por precedencia de operadores
- a. $\ast((\ast(x + 8)) + 28)$
b. $\ast((\ast(x + 2)) + 7)$
c. $(\ast\ast(x + 2) + 7)$
d. $\ast(\ast(x + 2) + 7)$
-
11. Justo antes de que una instrucción máquina escriba un resultado en memoria:
- a. en IR está el resultado y en MBR la dirección donde se almacenará
b. en IR está el resultado y en MAR la dirección donde se almacenará
c. en MAR está el resultado y en MBR la dirección donde se almacenará
d. en MBR está el resultado y en MAR la dirección donde se almacenará
-
12. En una arquitectura RISC típica:
- a. la UC es más compleja que en una arquitectura CISC
b. la programación resulta mucho más simple que en una arquitectura CISC
c. se usan pocas instrucciones de las disponibles en el conjunto de instrucciones
d. suele usarse segmentación
-
13. ¿Qué circuito suele utilizarse para traducir el código de operación de una instrucción máquina a dirección de comienzo en la memoria de control del microprograma correspondiente?
- a. **Una memoria**
b. Un multiplexor
c. Un contador
d. Un demultiplexor
-
14. ¿Cómo actúa el indicador de signo?
- a. **Se pone a 1 cuando el resultado es negativo**
b. Se pone a 1 cuando el resultado es distinto de cero
c. Se pone a 0 cuando el resultado es negativo

ENCENDER TU LLAMA CUESTA MUY POCO



- d. Se pone a 1 cuando el resultado es positivo
- 15.** ¿Cuál de las siguientes afirmaciones es cierta?
- La E/S en memoria emplea la patilla IO/M#
 - En E/S independiente, las instrucciones de acceso a memoria se emplean tanto para memoria como para E/S**
 - La E/S independiente facilita la protección
 - La E/S en memoria es mucho más rápida que la E/S independiente
- 16.** La técnica de sondeo, escrutinio o "polling"...
- Se utiliza para identificar la fuente de una interrupción**
 - No permite establecer un mecanismo de asignación de prioridades a los distintos dispositivos
 - En caso de utilizarse, es necesario emplear varias líneas para que los dispositivos soliciten una interrupción
 - Es incompatible con el daisy-chain
- 17.** Señale cuál de las siguientes opciones es una técnica habitual para llevar a cabo la transferencia de datos entre el computador y los dispositivos de E/S externos:
- E/S por nivel
 - E/S por flanco
 - Acceso directo a memoria (DMA)**
 - Acceso indirecto a memoria (IMA)
- 18.** ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es cierta?
- Es una bifurcación normalmente externa al programa en ejecución**
 - Su objetivo es incrementar el ancho de banda con el dispositivo
 - Solicita que el procesador se aisle de los buses
 - Permite realizar transferencias sin el control de un programa
- 19.** La primera instrucción ensamblador de una subrutina compilada con gcc en Linux/x86 cdecl suele ser:

- mov %esp, %ebp
- push %ebp**
- push %ebx
- pop %ebx

20. En cdecl/x86, ¿cuál de los siguientes registros tiene que ser guardado por la función llamada si es alterado por ésta?

- eax
- ebx**
- ecx
- edx

21. Se desea construir una memoria de SRAM de tamaño 3G X 8 a partir de elementos de memoria SRAM más pequeños. Cuál de las siguientes soluciones sería correcta:

- 256 chips de 16Mx 1 bits
- 16 chips de 512 M x 2 bits
- 12 chips de 512M x 4 bits**
- Ninguna de las anteriores es correcta

22. Un procesador de 1GHz tarda 4ns en realizar 4 instrucciones sin realizar segmentación de cauce. Cuanto tardaría en realizar 9 instrucciones un procesador con segmentación de cauce de 4 etapas si no existiera ningún retraso en ninguna de las instrucciones.

- 2 ns
- 3 ns**
- 4.5 ns
- 9 ns

23. ¿Cuántas patillas de dirección tiene una memoria DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?

- 20
- 16
- 30
- 15**

24. ¿Cuál de las siguientes afirmaciones es cierta?

- a. La memoria SRAM es más lenta que la DRAM

- b. La lectura en la memoria SRAM es destructiva

- c. La memoria DRAM es más cara que la SRAM

d. Ninguna de las anteriores

25. La ganancia en velocidad de un cauce de K etapas de igual duración ejecutando un programa de N instrucciones es:

a. $S = KN/(K-N+1)$

b. $S = NKT/(N-K+1)T$

c. **$S = KN/(K+N-1)$**

d. $S = NT/(N+K-1)T$

26. ¿A qué tipo de localidad de memoria hace referencia la siguiente afirmación: "si se referencia un elemento, tenderá a volver a ser referenciado pronto"?

- a. Localidad espacial

- b. Localidad lógica

c. **Localidad temporal**

- d. Ninguna de las respuestas anteriores es correcta
-

27. ¿Cuáles de las siguientes direcciones de memoria podrían estar simultáneamente en una memoria caché con correspondencia directa de 256 palabras con 16 palabras por bloque?

a. 0000 y FF0F

b. **ABAB y ABAC**

c. 08E3 y 74E1

d. Ninguna de las combinaciones anteriores

28. En una memoria DRAM que permite el acceso en modo página se accede a la palabra 0x1234. Si emplea páginas de 256 palabras, ¿Cuál será la menor dirección a la que podremos acceder rápidamente?

a. 0x1000

b. **0x1200**

c. 0x1230

d. Otra

29. ¿Cuál de las siguientes afirmaciones es cierta?

- a. Al realizar la segmentación de cauce aumenta en general el tiempo necesario para la ejecución de un programa
- b. Debido a que pueden existir dependencia de datos, los resultados de un programa pueden ser diferentes a si el programa se ejecutara sin segmentación
- c. La segmentación de cauce disminuye el número de instrucciones necesarias para la ejecución de un programa
- d. **Ninguna de las combinaciones anteriores**
-

30. En un procesador con segmentación de cauce, aumentar el número de etapas (p.ej. de 2 a 4, o de 4 a 8), tiene en general como consecuencia:

a. **Un incremento de las prestaciones**

- b. Un mayor retraso en la ejecución de los programas debido al incremento del número de etapas

- c. Una disminución en la posible dependencia de datos

- d. Una disminución de la máxima frecuencia de reloj a la que puede operar el cauce
-

A medida que nos acercamos a la CPU en una memoria organizada en forma jerárquica, ¿qué suele ocurrir con respecto al tamaño de las unidades de transferencia entre niveles?

Usuar Profesor

io es

- a) Aumenta
- b) Disminuye
- c) Permanece constante
- d) Todas las posibilidades anteriores suelen darse

Respecto al entrelazado de memoria:

Usuar Profesor

io es

- a) Empleando entrelazado de orden superior las direcciones de memoria se reparten de forma consecutiva dentro de un módulo
- b) El entrelazado de orden inferior permite el acceso simultáneo a palabras consecutivas
- c) La cantidad de conflictos en el acceso a la memoria suele ser menor en los sistemas que emplean entrelazado de orden inferior
- d) Todas las respuestas anteriores son ciertas

Una caché de 64 palabras y correspondencia por sectores utiliza bloques de una palabra. Si las direcciones de memoria principal son de 24 bits y cada sector está constituido por 32 bloques, el campo etiqueta debe tener:

Usuar Profesor

io es

- a) 19 bits
- b) 18 bits
- c) 20 bits
- d) 21 bits

¿Cuántas entradas suele tener un TLB?

Usuar Profesor

io es

- a) Tantas como número de páginas tenga la memoria física
- b) Muchas menos que el número de página de la memoria virtual
- c) Más que el número de páginas de la memoria virtual
- d) Una única entrada correspondiente a la última página accedida

¿Cuál de las siguientes características es típica de la microprogramación horizontal?

Usuar Profesor

io es

- a) Ninguna o escasa codificación.
- b) Microinstrucciones cortas.
- c) Escasa capacidad para expresar paralelismo entre microoperaciones.
- d) Muchos campos solapados.

¿Cuál de las siguientes afirmaciones sobre compilación C->ASM es falsa?

Usuar Profesor

io es

- a) Puede que el compilador elimine por optimización construcciones C enteras (como un bucle for completo), si se conoce el resultado en tiempo de compilación
- b) Puede que el compilador altere el orden del código C, apareciendo antes la traducción de sentencias C posteriores
- c) Puede suceder que varias sentencias C se traduzcan por una única instrucción ASM
- d) Puede que el compilador altere por optimización el orden de los parámetros pasados en una llamada a función, si el marco de pila resultante es más eficiente

En una arquitectura RISC típica:

Usuar Profesor

io es

- a) no puede usarse segmentación.
- b) la UC es más compleja que en una arquitectura CISC.

- c) la programación resulta mucho más simple que en una arquitectura CISC.
- d) se usan muchas instrucciones de las disponibles en el conjunto de instrucciones.

Puntuación: **1,00**

Un sistema de memoria utiliza una caché con bloques de 8 palabras y memorias DRAM trabajando en modo página. ¿Cuántos ciclos se necesitan para copiar un bloque de memoria en la caché?

Usuar Profesor

io es

- a) 4 ciclos completos RAS-CAS y 4 ciclos CAS
- b) 7 ciclos completos RAS-CAS y 1 ciclo CAS
- c) 8 ciclos completos RAS-CAS
- d) 1 ciclo completo RAS-CAS y 7 ciclos CAS

Puntuación: **-0,33**

Una memoria que está estructurada en palabras de 16 bits tiene una capacidad de 64 Kbits. ¿Cuántas palabras tiene?

Usuar Profesor

io es

- a) 4000
- b) 6400
- c) 4096
- d) 6553

Puntuación: **1,00**

¿En cuál de los siguientes casos es más adecuado utilizar DRAM que SRAM?

Usuar Profesor

io es

- a) Una memoria caché.
- b) Un sistema de memoria en el que el precio es el factor más importante.
- c) Un diseño en el que es importante que los datos se puedan almacenar durante una gran cantidad de tiempo sin que se realice ninguna acción por parte del procesador.

- d) Un sistema de memoria en el que las prestaciones son el objetivo más importante.

Puntuación: **1,00**

¿Cuántas líneas de dirección son necesarias en un memoria RAM de 256 K palabras dinámica? ¿Y estática?

Usuar Profesor

io es

- a) 9/9
- b) 18/1
8
- c) 18/9
- ✓** • d) 9/18

Puntuación: **1,00**

En una arquitectura RISC típica:

Usuar Profesor

io es

- ✓** • a) se usan muchas instrucciones de las disponibles en el conjunto de instrucciones.
- b) la programación resulta mucho más simple que en una arquitectura CISC.
- c) no puede usarse segmentación.
- d) la UC es más compleja que en una arquitectura CISC.

Puntuación: **1,00**

En una caché con 64 bytes de longitud de línea, ¿qué bits de una dirección de memoria se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección?

Usuar Profesor

io es

- a) Los 6 bits menos significativos
- b) Los 6 bits más significativos
- X** c) Los 4 bits menos significativos
- d) Los 4 bits más significativos

Puntuación: **-0,33**

[T6.5]

Un programa crea en memoria una larga secuencia de números de forma consecutiva. ¿Qué tipo de estrategia de mantenimiento de coherencia es más eficiente para ejecutar este programa en un sistema con jerarquía de memoria?

Usuar Profesor

io es

- ✓ •
 - a) Escritura directa ("write-through")
 - b) Post-escritura ("write-back")
 - c) Tanto a) como b) son igual de eficientes
 - d) No puede saberse qué técnica es mejor

Puntuación: **1,00**

[T6.3]

Un computador con 10 bits en el bus de direcciones puede direccionar como máximo:

Usuar Profesor

io es

- ✓ •
 - a) 65535 palabras
 - b) 1024 palabras
 - c) 1000 palabras
 - d) 65536 palabras

Puntuación: **1,00**

[T6.1]

Se dispone de un computador cuyo tiempo medio de acceso al sistema de memoria cache y memoria principal es de 18 ns. Si la tasa de fallo de la cache es de 0,2 y el tiempo de acceso a la memoria principal es 50 ns. ¿Cuál es el tiempo de acceso a la cache?

Usuar Profesor

io es

- - a) 8 ns
 - b) 24,4 ns
 - c) 10 ns
 - d) 6 ns

✗

Puntuación: **-0,33**

[T6.5]

Un computador emplea un sistema de memoria principal de 128 palabras y una memoria cache de 32 palabras. La organización de la memoria cache es totalmente asociativa y el tamaño de bloque es de 8 palabras. Se emplea el algoritmo de reemplazo FIFO. Si inicialmente la memoria cache está totalmente vacía, calcule el número de fallos cuando se lee la secuencia de direcciones de la memoria principal: 0000100, 1000001, 0000101, 0010011, 0100010, 1000100, 0000111.

Usuar Profesor

io es

- a) 6
fallos
- b) 5
fallos
- c) 3
fallos
- d) 4
fallos

Puntuación: **1,00**



Nombre:

DNI:

Grupo:

Test 8: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale $10/10 = 1,00$ puntos, 0 si no se contesta o está claramente tachada y $10/(3 \times 10) = -0,33$ si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

1	2	3	4	5	6	7	8	9	10

1. Se están probando varias implementaciones en ensamblador de la función factorial. ¿Cuál **no debería funcionar**?

```
unsigned f(unsigned n)
{
    if (n < 3)
        return n;
    else
        return n * f(n - 1);
}

a) unsigned total = 1;
asm("1: imul %1, %0 \n"
    "    loop 1b      \n"
    :"+r"(total):"c"(n));
NOOO      return total;

b) Ninguna otra respuesta es correcta.
c) unsigned total = 1;
while (n)
    asm("imul %1, %0 \n"
        "dec %1          \n"
        :"+r"(total), "+r"(n));
return total;
d) unsigned total = 1;
while (n)
    asm("imul %1, %0 \n"
        "dec %1          \n"
        :"+r"(total):"r"(n));
return total;
```

2. ¿Qué registros podemos modificar a nuestro antojo en el interior de una función que siga el protocolo “System V AMD64 ABI”, también llamados **salva invocante**?

- a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
b) Ninguna otra respuesta es correcta.

- c) RBX, RBP, R12, R13, R14 y R15.
d) RSP.
3. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?
- a) En la pila.
b) En la memoria.
c) Ninguna otra respuesta es correcta.
d) En registros.

4. ¿Cuál es la forma correcta de utilizar la instrucción “popcnt” para calcular el número de bits a 1 en un entero sin signo de 32 bits dentro de la siguiente función?

```
unsigned f(unsigned u)
{
    unsigned c;
    // ...
    return c;
}
```

- a) asm("popcnt %1, %0": "=r"(c):"r"(u));
b) asm("popcnt %0, %1": "=r"(c), "=r"(u));
c) Ninguna otra respuesta es correcta.
d) asm("popcnt %0, %1": "+r"(c):"r"(u));

5. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp    %esi,%edi
mov    %esi,%eax
cmovle %edi,%eax
ret
```

- a) int f(int a, int b)
{
 if (a < b) return a;

- ```

 else return b;
}

b) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

c) Ninguna otra respuesta es correcta.

d) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

6. ¿A qué función puede corresponder el siguiente código ensamblador?

mov %esi,%eax
ret

a) Ninguna otra respuesta es correcta.

b) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

c) int f(int a, int b)
{
 if (a < b) return a;
 else return b;
}

d) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

7. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo

```

lo “System V AMD64 ABI”, también llamados **salva invocado**?

- a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
- b) Ninguna otra respuesta es correcta.
- c) RBX, RBP, R12, R13, R14 y R15.
- d) RSP.

8. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

- a) RCX    b) RSP    c) RAX    d) RBX

9. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov %esi,%eax
ret
```

- a) `return a;`
- b) Ninguna otra respuesta es correcta.
- c) `return c;`
- d) `return b;`

10. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov %edi,%eax
ret
```

- a) `return c;`
- b) `return a;`
- c) Ninguna otra respuesta es correcta.
- d) `return b;`



Nombre:

DNI:

Grupo:

### Test 13: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale  $10/10 = 1,00$  puntos, 0 si no se contesta o está claramente tachada y  $10/(3 \times 10) = -0,33$  si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
|   |   |   |   |   |   |   |   |   |    |

1. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov %esi,%eax
ret
```

- a) `return c;`
- b) `return b;`**
- c) Ninguna otra respuesta es correcta.
- d) `return a;`

2. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

- a) RBX
- b) RCX
- c) RAX**
- d) RSP

3. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo “System V AMD64 ABI”, también llamados **salva invocado**?

SI

NO

- a) RSP.**
- b) RBX, RBP, R12, R13, R14 y R15.**
- c) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.**
- d) Ninguna otra respuesta es correcta.

4. ¿Cuál es la forma correcta de utilizar la instrucción “popcnt” para calcular el número de bits a 1 en un entero sin signo de 32 bits dentro de la siguiente función?

```
unsigned f(unsigned u)
{
 unsigned c;
 // ...
 return c;
}
```

**a) `asm("popcnt %1, %0": "=r"(c) :"r"(u));`**

b) Ninguna otra respuesta es correcta.

c) `asm("popcnt %0, %1": "+r"(c) :"r"(u));`

d) `asm("popcnt %0, %1": "=r"(c), "=r"(u));`

5. ¿Cuál es la forma correcta de utilizar la instrucción “lea” para implementar la función `f` mediante una única línea en ensamblador en línea?

```
int f(int a, int b)
{
 return 2 * a + b;
}
```

**a) `asm("lea (%1,%0,2), %0": "=a"(a) :"r"(b));`**

**b) `asm("lea (%1,%0,2), %0": "+a"(a) :"r"(b));`**

c) `asm("lea (%1,%0,2), %0": "+a"(b) :"r"(a));`

d) Ninguna otra respuesta es correcta.

6. Dado el siguiente código C y su implementación en ensamblador, ¿Cuánto vale N?

```
int f(int a, int b)
{
 int c = 7;
 for (int i = 0; i < N; ++i)
 c += a * i + b;
 return c;
}
```

```
lea 0x7(%rdi,%rsi,2),%edx
lea (%rsi,%rdi,2),%eax
add %edx,%eax
ret
```

SI      NO

- a) 3**
- b) 2**
- c) 0
- d) 1

7. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?

- a) En registros.**

b) Ninguna otra respuesta es correcta.

c) En la pila.

d) En la memoria.

8. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp %esi,%edi
mov %esi,%eax
cmovge %edi,%eax
ret
```

a) int f(int a, int b)  
{  
 if (a == b) return a;  
 else return b;  
}

b) int f(int a, int b)  
{  
 if (a < b) return a;  
 else return b;  
}

c) int f(int a, int b)  
{  
 if (a > b) return a;  
 else return b;  
}

d) Ninguna otra respuesta es correcta.

9. ¿Cuál será el código de una función cuya definición es int f(int a, int b, int c) y que se implementa así?

```
mov %edx,%eax
ret
```

a) return a;

b) return c;

c) Ninguna otra respuesta es correcta.

d) return b;

10. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp %esi,%edi
mov %esi,%eax
cmovle %edi,%eax
ret
```

a) int f(int a, int b)  
{  
 if (a == b) return a;  
 else return b;  
}

b) Ninguna otra respuesta es correcta.

c) int f(int a, int b)  
{  
 if (a < b) return a;  
 else return b;  
}

d) int f(int a, int b)  
{  
 if (a > b) return a;  
 else return b;  
}

Nombre:

DNI:

Grupo:

### Test 14: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale  $10/10 = 1,00$  puntos, 0 si no se contesta o está claramente tachada y  $10/(3 \times 10) = -0,33$  si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

|   |   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|---|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|   |   |   |   |   |   |   |   |   |    |

1. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov %esi,%eax
ret
```

- a) `return b;`
- b) `return a;`
- c) `return c;`
- d) Ninguna otra respuesta es correcta.

2. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo “System V AMD64 ABI”, también llamados **salva invocado**?

- a) Ninguna otra respuesta es correcta.
- b) RSP.
- c) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
- d) RBX, RBP, R12, R13, R14 y R15.

3. Se están probando varias implementaciones en ensamblador de la función factorial. ¿Cuál **no debería funcionar**?

```
unsigned f(unsigned n)
{
 if (n < 3)
 return n;
 else
 return n * f(n - 1);
}
```

- a) `unsigned total = 1;
while (n)
 asm("imul %1, %0 \n"
 "dec %1 \n"
 :"+r"(total):"r"(n));
return total;`

```
b) unsigned total = 1;
while (n)
 asm("imul %1, %0 \n"
 "dec %1 \n"
 :"+r"(total),"+r"(n));
return total;
```

- c) `unsigned total = 1;
asm("1: imul %1, %0 \n"
 "loop 1b \n"
 :"+r"(total):"c"(n));
return total;`
- d) Ninguna otra respuesta es correcta.

4. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?

- a) En la memoria.
- b) `En registros.`
- c) En la pila.
- d) Ninguna otra respuesta es correcta.

5. Dado el siguiente código C y su implementación en ensamblador, ¿Cuánto vale N?

```
int f(int a, int b)
{
 int c = 7;
 for (int i = 0; i < N; ++i)
 c += a * i + b;
 return c;
}
lea 0x7(%rdi,%rsi,2),%edx
lea (%rsi,%rdi,2),%eax
add %edx,%eax
ret
```

- a) 3
- b) 2
- c) 1
- d) 0

6. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp %esi,%edi
mov %esi,%eax
cmovle %edi,%eax
ret
```

a) int f(int a, int b)  
{  
 if (a < b) return a;  
 else return b;  
}

b) Ninguna otra respuesta es correcta.

c) int f(int a, int b)  
{  
 if (a > b) return a;  
 else return b;  
}

d) int f(int a, int b)  
{  
 if (a == b) return a;  
 else return b;  
}

7. ¿Cuál será el código de una función cuya definición es int f(int a, int b, int c) y que se implementa así?

```
mov %edi,%eax
ret
```

a) Ninguna otra respuesta es correcta.

b) return a;

c) return b;

d) return c;

8. ¿Qué registros podemos modificar a nuestro antojo en el interior de una función que sigua el protocolo

lo “System V AMD64 ABI”, también llamados **salva invocante**?

a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.

b) Ninguna otra respuesta es correcta.

c) RBX, RBP, R12, R13, R14 y R15.

d) RSP.

9. ¿A qué función puede corresponder el siguiente código ensamblador?

```
mov %esi,%eax
ret
```

a) int f(int a, int b)  
{  
 if (a < b) return a;  
 else return b;  
}

b) int f(int a, int b)  
{  
 if (a > b) return a;  
 else return b;  
}

c) Ninguna otra respuesta es correcta.

d) int f(int a, int b)  
{  
 if (a == b) return a;  
 else return b;  
}

10. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

a) RAX    b) RSP    c) RCX    d) RBX

**Test 13: 10 puntos.**

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale  $10/10 = 1,00$  puntos, 0 si no se contesta o está claramente tachada y  $10/(3 \times 10) = -0,33$  si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

|   |   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|---|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| D | A | C | D | B | B | A | B | A | A  |

1. ¿Qué tipo de llamada hace el siguiente fragmento de código?

```
xor %edi, %edi
call exit
```

- a) una llamada a un controlador de interrupción
- b) una llamada a función
- c) Ninguna otra respuesta es correcta
- d) una llamada al sistema**

2. ¿Qué se obtiene al intentar observar en memoria un objeto de 8 bytes declarado “resultado .quad 0 x8877665544332211” mediante la orden de gdb “x /4hx &resultado”? Nota: la dirección de memoria puede variar.

- a) 0x403020: 0x2211 0x4433 0x6655 0x8877**
- b) 0x403020: 0x1122 0x3344 0x5566 0x7788
- c) 0x403020: 0x8877 0x6655 0x4433 0x2211
- d) 0x403020: 0x7788 0x5566 0x3344 0x1122

3. ¿De qué tipo dice “file” que son los ficheros obtenidos al ejecutar “gcc -S msum.c sum.c”?

- a) ELF 64-bit LSB relocatable, x86-64, version 1 (SYSV), not stripped
- b) ELF 64-bit LSB executable, x86-64, version 1 (SYSV)...
- c) assembler source, ASCII text**
- d) C source, ASCII text

4. ¿Qué devuelve cuando acaba bien la función puts de la biblioteca de C?

- a) Ninguna otra respuesta es correcta
- b) un número no negativo
- c) EOF

- d) el número de caracteres impresos**

5. ¿Qué función codifica el siguiente fragmento de código?

```
0: cmp %esi,%edi
2: mov %esi,%eax
4: cmovge %edi,%eax
7: retq
```

```
a) int min(int a, int b)
{
 if (a < b)
 return a;
 else
 return b;
}

b) int max(int a, int b)
{
 if (a > b)
 return a;
 else
 return b;
}

c) bool distinto(int a, int b)
{
 if (a != b)
 return true;
 else
 return false;
}
```

```
d) bool igual(int a, int b)
{
 if (a == b)
 return true;
 else
 return false;
}
```

6. ¿Qué tipo de llamada hace el siguiente fragmento

de código?

```
mov $60, %rax
xor %rdi, %rdi
syscall
```

- a) una llamada a un controlador de interrupción
  - b) una llamada al sistema**
  - c) Ninguna otra respuesta es correcta
  - d) una llamada a función
7. ¿Qué código cree mejor para copiar en edx:eax un valor de 64 bits almacenado en el tope de la pila sin alterarla?

- a) mov (%rsp), %eax  
 mov 4(%rsp), %edx**
- b) mov (%rsp), %rax  
 mov %rax, %rdx  
 and \$0xffffffff, %rax  
 shr \$32, %rdx
- c) pop %rax  
 mov %rax, %rdx  
 and \$0xffffffff, %rax  
 shr \$32, %rdx
- d) movw (%rsp), %ax  
 movw 4(%rsp), %dx

8. ¿Qué código utilizaría para conseguir que un número entero contenido en el registro eax se **expanda** correctamente a la pareja de registros edx:eax?

- a) cltd
- b) cltq**
- c) Ninguna otra respuesta es correcta
- d) mov %edx, %eax  
 sar \$31, %eax

9. ¿Cuál de las siguientes formas de finalizar un programa **no es correcta**?

- a) Ninguna otra respuesta es correcta**

- b) mov \$60, %rax  
 mov \$0, %rdi  
 syscall
- c) mov \$0, %rdi  
 call exit
- d) xor %rdi, %rdi  
 call exit

10. ¿Qué función codifica el siguiente fragmento de código?

```
0: cmp %esi,%edi
2: sete %al
5: retq
```

- a) bool igual(int a, int b)**  
{  
 if (a == b)  
 return true;  
 else  
 return false;  
}  
  
b) int max(int a, int b)  
{  
 if (a > b)  
 return a;  
 else  
 return b;  
}  
  
c) int min(int a, int b)  
{  
 if (a < b)  
 return a;  
 else  
 return b;  
}  
  
d) bool distinto(int a, int b)  
{  
 if (a != b)  
 return true;  
 else  
 return false;  
}



Nombre:

DNI:

Grupo:

### Test 8: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale  $10/10 = 1,00$  puntos, 0 si no se contesta o está claramente tachada y  $10/(3 \times 10) = -0,33$  si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|----|
|   |   |   |   |   |   |   |   |   |    |

1. Se están probando varias implementaciones en ensamblador de la función factorial. ¿Cuál **no debería funcionar**?

```
unsigned f(unsigned n)
{
 if (n < 3)
 return n;
 else
 return n * f(n - 1);
}

a) unsigned total = 1;
asm("1: imul %1, %0 \n"
 " loop 1b \n"
 :"+r"(total):"c"(n));
NOOO return total;

b) Ninguna otra respuesta es correcta.
c) unsigned total = 1;
while (n)
 asm("imul %1, %0 \n"
 "dec %1 \n"
 :"+r"(total), "+r"(n));
return total;
d) unsigned total = 1;
while (n)
 asm("imul %1, %0 \n"
 "dec %1 \n"
 :"+r"(total):"r"(n));
return total;
```

2. ¿Qué registros podemos modificar a nuestro antojo en el interior de una función que sigua el protocolo “System V AMD64 ABI”, también llamados **salva invocante**?

a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.  
b) Ninguna otra respuesta es correcta.

- c) RBX, RBP, R12, R13, R14 y R15.  
d) RSP.
3. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?
- a) En la pila.  
b) En la memoria.  
c) Ninguna otra respuesta es correcta.  
d) En registros.

4. ¿Cuál es la forma correcta de utilizar la instrucción “popcnt” para calcular el número de bits a 1 en un entero sin signo de 32 bits dentro de la siguiente función?

```
unsigned f(unsigned u)
{
 unsigned c;
 // ...
 return c;
}
```

- a) asm("popcnt %1, %0": "=r"(c):"r"(u));
b) asm("popcnt %0, %1": "=r"(c), "=r"(u));
c) Ninguna otra respuesta es correcta.
d) asm("popcnt %0, %1": "+r"(c):"r"(u));

5. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp %esi,%edi
mov %esi,%eax
cmovle %edi,%eax
ret
```

- a) int f(int a, int b)
{
 if (a < b) return a;

- ```

    else      return b;
}

b) int f(int a, int b)
{
    if (a == b) return a;
    else      return b;
}

c) Ninguna otra respuesta es correcta.

d) int f(int a, int b)
{
    if (a > b) return a;
    else      return b;
}

6. ¿A qué función puede corresponder el siguiente código ensamblador?

mov    %esi,%eax
ret

a) Ninguna otra respuesta es correcta.

b) int f(int a, int b)
{
    if (a == b) return a;
    else      return b;
}

c) int f(int a, int b)
{
    if (a < b) return a;
    else      return b;
}

d) int f(int a, int b)
{
    if (a > b) return a;
    else      return b;
}

7. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo

```

lo “System V AMD64 ABI”, también llamados **salva invocado**?

- a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
- b) Ninguna otra respuesta es correcta.
- c) RBX, RBP, R12, R13, R14 y R15.
- d) RSP.

8. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

- a) RCX b) RSP c) RAX d) RBX

9. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov    %esi,%eax
ret
```

- a) `return a;`
- b) Ninguna otra respuesta es correcta.
- c) `return c;`
- d) `return b;`

10. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov    %edi,%eax
ret
```

- a) `return c;`
- b) `return a;`
- c) Ninguna otra respuesta es correcta.
- d) `return b;`



Nombre:

DNI:

Grupo:

Test 13: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale $10/10 = 1,00$ puntos, 0 si no se contesta o está claramente tachada y $10/(3 \times 10) = -0,33$ si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

1	2	3	4	5	6	7	8	9	10

1. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov    %esi,%eax
ret
```

- a) `return c;`
b) `return b;`
c) Ninguna otra respuesta es correcta.
d) `return a;`

2. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

a) RBX b) RCX c) RAX d) RSP

3. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo “System V AMD64 ABI”, también llamados **salva invocado**?

SI

NO

a) RSP.
b) RBX, RBP, R12, R13, R14 y R15.
c) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
d) Ninguna otra respuesta es correcta.

4. ¿Cuál es la forma correcta de utilizar la instrucción “popcnt” para calcular el número de bits a 1 en un entero sin signo de 32 bits dentro de la siguiente función?

```
unsigned f(unsigned u)
{
    unsigned c;
    // ...
    return c;
}
```

- a) `asm("popcnt %1, %0": "=r"(c) : "r"(u));`
b) Ninguna otra respuesta es correcta.
c) `asm("popcnt %0, %1": "+r"(c) : "r"(u));`
d) `asm("popcnt %0, %1": "=r"(c), "=r"(u));`

5. ¿Cuál es la forma correcta de utilizar la instrucción “lea” para implementar la función f mediante una única línea en ensamblador en línea?

```
int f(int a, int b)
{
    return 2 * a + b;
}
```

- a) `asm("lea (%1,%0,2), %0": "=a"(a) : "r"(b));`
b) `asm("lea (%1,%0,2), %0": "+a"(a) : "r"(b));`
c) `asm("lea (%1,%0,2), %0": "+a"(b) : "r"(a));`
d) Ninguna otra respuesta es correcta.

6. Dado el siguiente código C y su implementación en ensamblador, ¿Cuánto vale N?

```
int f(int a, int b)
{
    int c = 7;
    for (int i = 0; i < N; ++i)
        c += a * i + b;
    return c;
}
```

```
lea    0x7(%rdi,%rsi,2),%edx
lea    (%rsi,%rdi,2),%eax
add   %edx,%eax
ret
```

- SI NO
a) 3 b) 2 c) 0 d) 1

7. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?

- a) En registros.

b) Ninguna otra respuesta es correcta.

c) En la pila.

d) En la memoria.

8. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp    %esi,%edi  
mov    %esi,%eax  
cmovge %edi,%eax  
ret
```

a) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

b) int f(int a, int b)
{
 if (a < b) return a;
 else return b;
}

c) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

d) Ninguna otra respuesta es correcta.

9. ¿Cuál será el código de una función cuya definición es int f(int a, int b, int c) y que se implementa así?

```
mov    %edx,%eax  
ret
```

a) return a;

b) return c;

c) Ninguna otra respuesta es correcta.

d) return b;

10. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp    %esi,%edi  
mov    %esi,%eax  
cmovle %edi,%eax  
ret
```

a) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

b) Ninguna otra respuesta es correcta.

c) int f(int a, int b)
{
 if (a < b) return a;
 else return b;
}

d) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

Nombre:

DNI:

Grupo:

Test 14: 10 puntos.

Escriba la opción correcta dentro de la casilla debajo de cada número de pregunta. Cada respuesta correcta vale $10/10 = 1,00$ puntos, 0 si no se contesta o está claramente tachada y $10/(3 \times 10) = -0,33$ si es errónea o no está claramente contestada. Se aconseja terminar de leer completamente cada pregunta antes de contestarla.

1	2	3	4	5	6	7	8	9	10

1. ¿Cuál será el código de una función cuya definición es `int f(int a, int b, int c)` y que se implementa así?

```
mov    %esi,%eax
ret
```

- a) `return b;`
- b) `return a;`
- c) `return c;`
- d) Ninguna otra respuesta es correcta.

2. ¿Qué registros estamos obligados a preservar en el interior de una función que sigue el protocolo “System V AMD64 ABI”, también llamados **salva invocado**?

- a) Ninguna otra respuesta es correcta.
- b) RSP.
- c) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.
- d) RBX, RBP, R12, R13, R14 y R15.

3. Se están probando varias implementaciones en ensamblador de la función factorial. ¿Cuál **no debería funcionar**?

```
unsigned f(unsigned n)
{
    if (n < 3)
        return n;
    else
        return n * f(n - 1);
}
```

- a) `unsigned total = 1;
while (n)
 asm("imul %1, %0 \n"
 "dec %1 \n"
 :"+r"(total):"r"(n));
return total;`

```
b) unsigned total = 1;
while (n)
    asm("imul %1, %0 \n"
        "dec %1      \n"
        :"+r"(total),"+r"(n));
return total;
```

- c) `unsigned total = 1;
asm("1: imul %1, %0 \n"
 "loop 1b \n"
 :"+r"(total):"c"(n));
return total;`
- d) Ninguna otra respuesta es correcta.

4. ¿Dónde se pasan los primeros 6 parámetros a una función mediante el protocolo “System V AMD64 ABI”?

- a) En la memoria.
- b) `En registros.`
- c) En la pila.
- d) Ninguna otra respuesta es correcta.

5. Dado el siguiente código C y su implementación en ensamblador, ¿Cuánto vale N?

```
int f(int a, int b)
{
    int c = 7;
    for (int i = 0; i < N; ++i)
        c += a * i + b;
    return c;
}
lea    0x7(%rdi,%rsi,2),%edx
lea    (%rsi,%rdi,2),%eax
add   %edx,%eax
ret
```

- a) 3
- b) 2
- c) 1
- d) 0

6. ¿A qué función puede corresponder el siguiente código ensamblador?

```
cmp    %esi,%edi  
mov    %esi,%eax  
cmovle %edi,%eax  
ret
```

a) int f(int a, int b)
{
 if (a < b) return a;
 else return b;
}

b) Ninguna otra respuesta es correcta.

c) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

d) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

7. ¿Cuál será el código de una función cuya definición es int f(int a, int b, int c) y que se implementa así?

```
mov    %edi,%eax  
ret
```

a) Ninguna otra respuesta es correcta.

b) return a;

c) return b;

d) return c;

8. ¿Qué registros podemos modificar a nuestro antojo en el interior de una función que sigua el protocolo

lo “System V AMD64 ABI”, también llamados **salva invocante**?

a) RAX, RCX, RDX, RDI, RSI, R8, R9, R10 y R11.

b) Ninguna otra respuesta es correcta.

c) RBX, RBP, R12, R13, R14 y R15.

d) RSP.

9. ¿A qué función puede corresponder el siguiente código ensamblador?

```
mov    %esi,%eax  
ret
```

a) int f(int a, int b)
{
 if (a < b) return a;
 else return b;
}

b) int f(int a, int b)
{
 if (a > b) return a;
 else return b;
}

c) Ninguna otra respuesta es correcta.

d) int f(int a, int b)
{
 if (a == b) return a;
 else return b;
}

10. ¿En qué registro devuelven el resultado las funciones que siguen la convención de llamada a función “System V AMD64 ABI”?

a) RAX b) RSP c) RCX d) RBX

1. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción no es cierta?
 - a) Es una bifurcación normalmente externa al programa en ejecución
 - b) Su objetivo es reclamar la atención del procesador
 - c) Solicita que se ejecute un programa específico para tratarla
 - d) Ninguna de las anteriores
2. Un procesador con una unidad de control microprogramada tiene una memoria de control de 256 palabras de 16 bits, de las que 128 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?
 - a) No se producir ahorro
 - b) 3840 bits
 - c) 256 bits
 - d) 4096 bits
3. ¿Cuál de las siguientes listas está correctamente ordenada temporalmente?
 - a) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX
486 posterior a 8086
 - b) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
486 posterior a 8086
 - c) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
 - d) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
Pentium III posterior a Pentium MMX
4. ¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:

```
push #4
push #7
push #8
add
push #10
sub
mul
```

 - a) 4, 7, 8, 10
 - b) 4, 7, 48
 - c) 20
 - d) 4

5. Si el registro `rax` contiene un long (64 bits CON signo) `x`, la secuencia de instrucciones siguiente:

```
cmpq $10, %rax  
ja dest
```

saltará a la etiqueta `dest` si:

- a) $x > 9$
- b) $x < 10$
- c) $x \leq -1 \text{ || } x \geq 11$
- d) $x \geq 0 \text{ && } x \leq 10$

6. ¿Cuál de los siguientes microprocesadores no es de 64 bits?

- a) Itanium
 - los 64bits "primeros" de Intel
- b) Core i7
- c) Core 2
 - posterior a los Pentium
- d) Pentium III
 - anterior a Pentium 4 e año 2004 primer EM64T

7. ¿Cuántos controladores de interrupciones 8259 hacen falta como mínimo para manejar 25 líneas de interrupción?

- a) 4
- b) 3
- c) 2
- d) 1

8. El Intel 8086:

- a) Fue el primer microprocesador.
 - de Intel de 16 bit
- b) Incluía instrucciones de multiplicación.
 - mul e imul, en concreto
- c) Contaba con 29 millones de transistores.
 - 29K, no 29M
- d) Podía direccionar 1 GB.
 - 1MB, no 1GB

9. Un fragmento del desensamblado de una “bomba” similar a las estudiadas en prácticas es:

```
400634: lea 0x4(%rsp),%rsi
400639: lea 0x1a4(%rip),%rdi
          # 4007e4 <_stdin+0xe4>
400640: mov $0x0,%eax
400645: callq 4004c0 <scanf>
40064a: mov 0x4(%rsp),%eax
40064e: shr %eax
400650: jc 40066a <L6>
400652: shr %eax
400654: jnc 40066a <L6>
400656: shr %eax
400658: jc 40066a <L6>
40065a: shr %eax
40065c: jnc 40066a <L6>
40065e: shr %eax
400660: jc 40066a <L6>
400662: shr %eax
400664: jnc 40066a <L6>
400666: test %eax,%eax
400668: jz 40066f <L7>
40066a: callq 4005c7 <boom>
40066f: callq 4005eb <defused>
```

El código numérico correcto (pin) es...

- a) 42
- b) el entero 0x40066A
- c) el entero almacenado a partir de 0x4(%rsp)
- d) ninguno de los anteriores

10. Respecto a las unidades de control nanoprogramadas:

- a) El diseño de las unidades de control nanoprogramadas debe ser vertical.
- b) La anchura de la memoria de nanoprograma es la misma que la de memoria de microprograma en un diseño de la misma unidad de control que no usara nanoprogramación.
- c) La realización nanoprogramada de una unidad de control es más rápida que la microprogramada.
- d) Suponiendo una memoria de microprograma con n microinstrucciones de w bits cada una, de las cuales 2^m son distintas, el ahorro en bits si se utiliza nanoprogramación es $(n \cdot m + 2^m \cdot w) - n \cdot w$.

11. Una máquina superescalar es aquella que:
- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
 - b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
 - c) emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.
 - d) ninguna respuesta de las anteriores es correcta.
12. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción no es cierta?
- a) Es una bifurcación normalmente externa al programa en ejecución
 - b) Su objetivo es reclamar la atención del procesador
 - c) Sigue la ejecución de un programa específico para tratarla
 - d) Ninguna de las anteriores
13. ¿Cuál de los siguientes elementos no forma parte de un canal de un controlador de acceso directo a memoria?
- a) Registro de dirección.
 - b) Registro de prioridades.
- En tr.8.2 figura FUERA de los canales, porque la prioridad sería cuando hay VARIOS canales, no tiene sentido hablar de prioridad DENTRO de un canal
- c) Registro contador.
 - d) Todos los elementos anteriores forman parte de un canal de DMA.
14. En la práctica de la bomba, el segundo ejercicio consistía en crear un ejecutable sin “explosiones”, para lo cual se puede utilizar... (marcar la opción *falsa*)
- a) gdb
 - b) ddd
 - c) hexedit
 - d) objdump
15. El circuito que hay que montar en la práctica del zumbador pasivo con Arduino es un:
- a) Buffer triestado
 - b) Divisor de tensión
 - c) Seguidor de voltaje
 - d) Comparador

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

16. Si el acceso directo a memoria se realiza mediante robo de ciclo:
- a) es posible que la ejecución de una operación elemental de transferencia en el bus sea temporalmente detenida
 - b) es posible que la ejecución de una instrucción máquina sea temporalmente detenida**
 - c) es necesario que termine de ejecutarse la instrucción máquina actual para comenzar una transferencia por DMA
 - d) ninguna de las anteriores es cierta
17. ¿Cuál de las siguientes afirmaciones acerca de las interrupciones en el PC (modo real) es cierta?
- a) Existen 1024 vectores de interrupción.
 - b) vector de interrupción es una doble palabra de 32 bits formada en primer lugar (dirección menor) por el segmento y seguida por el desplazamiento (dirección mayor) de cada rutina de servicio de interrupción
 - c) No todas las interrupciones se pueden generar por software
 - d) Ninguna de las anteriores afirmaciones es cierta**
18. En una unidad de control microprogramada se tienen dos campos de 5 señales de control mutuamente exclusivos, de manera que nunca se activarían señales de ambos campos en el mismo ciclo de reloj. Con esas 10 señales sería entonces posible...
- a) codificarlas con 4 bits, y sobraría un código que quedaría sin uso
 - b) codificarlas con 5 bits, y no sobraría ningún código sin uso
 - c) solaparlas en un solo campo de 5 bits, ahorrando por tanto 4 bits**
 - d) solaparlas en un solo campo de 6 bits, ahorrando por tanto 5 bits
19. En los casos concretos indicados para las siguientes instrucciones x86-64, ¿cuál no funciona como instrucción de transferencia?
- a) `pushq %rax`
transfiere RAX a la pila
 - b) `movq $0x15, %rax`
Ajusta RAX a 15 (transfiere el valor inmediato)
 - c) `cmpq 0x6000f0, %rax`**
No mueve datos, sólo ajusta flags, no es transferencia, es op.aritmética
 - d) `leaq variable, %rax`
Ajusta RAX a 0x6000... (transfiere la dirección de la variable)

20. Un computador con 15 líneas de direcciones tiene 3 módulos de memoria de 2^{13} palabras y utiliza E/S mapeada en memoria. ¿Cuál es el número máximo de periféricos que pueden conectarse, si cada uno de ellos utiliza 8 direcciones?

- a) 2^{10}
- b) 2^{13}
- c) 2^{11}
- d) 2^{12}

21. ¿Cuál de las siguientes instrucciones máquina copia en RAX la dirección efectiva resultante de la operación $RDX*8 + RBX$?

- a) `leaq (%rbx, %rdx, 8), %rax`

lea calcula la dirección efectiva $[RBX+RDX*8]$ y la almacena en RAX
También podría hacerlo con direcciones 32b, pero no lo hemos visto en clase

Se ha preferido cambiar a registros 64bits, factor escala 8x

En la pregunta de examen original (Jul'17) se mencionaban regs.32b, factor 4x

- b) `movq (%rbx, %rdx, 8), %rax`
mov mueve desde memoria
- c) `leaq 8(%rdx, %rdx), %rax`
la dirección efectiva sería $8+2*RDX$
- d) `movq 8(%rdx, %rdx), %rax`
mov mueve desde memoria, y además la dirección efectiva sería $8+2*RDX$

22. La extensión de signo a m bits de un número original N de n bits, con $m > n$, consiste en:

- a) Realizar la operación $2^m - N$
- b) Realizar la operación $2^m - N - 1$
- c) Incrementar la cantidad de bits a m preservando el signo y el valor del número.
- d) Incrementar la cantidad de bits a m rellenando con unos por la izquierda.

23. Al traducir de lenguaje C a ensamblador, gcc en máquinas Linux/x86-64 almacena (reserva espacio para) una variable “var” local a una función “fun” en una dirección de memoria referenciable (en lenguaje ensamblador) como:
- a) var (el nombre de la variable representa su posición de memoria)
no, eso sería para variables globales con gcc -m32 (ahora gcc viene con --enable-default-pie)
 - b) var(%rip)
eso sería una variable global (en Position-Independent-Executable)
 - c) k(%rsp), siendo k un número entero relativamente pequeño
sí, positivo (normal) o negativo (zona roja), aunque no hayamos visto zona roja en CS:APP3e
 - d) k(%rbp), siendo k un entero positivo relativamente pequeño
no, ni siquiera cuando gcc usara puntero marco (por tamaño variable del marco, ver CS:APP3e 3.10.5 p.326-329) ni si opcionalmente se compilara con -fno-omit-frame-pointer, porque en todo caso sería -k(%rbp)
24. En x86-64, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?
- a) mov \$-1, %edi
mov no afecta a los flags, los demás sí: no hace falta comprobar más calculamos los flags por curiosidad, no porque haga falta para responder
 - b) xor %edi, %edi
add \$-1, %edi
 - c) sub %edi, %edi
adc \$0xFFFFFFFF, %edi
sub/adc afectan: CZSO=0100 -> 0010
 - d) mov \$-1, %edi
add \$0, %edi
add afecta: CZSO=0010
25. ¿Cuál de las siguientes afirmaciones es correcta?
- a) El lenguaje máquina es igual para todos los computadores.
 - b) Las instrucciones en lenguaje máquina se almacenan y tratan como cadenas de unos y ceros.
 - c) El lenguaje ensamblador es igual para todos los computadores.
 - d) Las instrucciones en lenguaje ensamblador se almacenan y tratan como cadenas de unos y ceros.

26. Si el contenido de RBP es 0x13000, ¿a qué dirección se hace referencia con la instrucción INC –0x80(%RBP)?
- a) 0x12F80
 - b) 0x80
 - c) 0x13080
 - d) 0x13000
27. La técnica de Consulta de Estado (polling) puede usarse para...
- a) identificar el origen de una interrupción
 - b) consultar si el dispositivo está dispuesto para entregar o recibir datos
 - c) consultar el sentido del DMA en curso (desde memoria / hacia memoria)
 - d) establecer un mecanismo software de asignación de prioridades a los dispositivos
28. Suponiendo que varios dispositivos comparten una única línea de solicitud de interrupción y que varios de ellos solicitan una interrupción al mismo tiempo, ¿qué dispositivo tendría mayor prioridad a la hora de ser atendidas sus peticiones?
- a) Si se emplea una técnica de sondeo ("polling"), el dispositivo cuyo estado se consulte primero
 - b) Si se emplea una técnica de encadenamiento ("daisy-chain"), el dispositivo al que primero se conecta la línea INTA# del procesador
 - c) Las respuestas a y b son ciertas
 - d) Las respuestas a y b son falsas
29. Al colocar un led en la placa de prototipado de Arduino, ¿cómo se sabe cuáles son el ánodo y el cátodo?
- a) Mirando las leyendas A+ o K- impresas en las patillas del led
 - b) No es necesario saberlo, la polaridad es indiferente a la hora de colocar un led en un circuito
 - c) Mirando los códigos de color dibujados sobre el encapsulado
 - d) Mirando la longitud de las patillas o qué lado del encapsulado es plano
30. Respecto a registros base e índice en x86-64, la excepción es que
- a) RBP no puede ser registro base
 - b) RBP no puede ser registro índice
 - c) RSP no puede ser registro base
 - d) RSP no puede ser registro índice
31. La microprogramación vertical se caracteriza por tener:
- a) microinstrucciones largas
 - b) escaso o ningún solapamiento entre campos
 - c) capacidad para expresar un alto grado de paralelismo en las microoperaciones a ejecutar
 - d) mucha codificación

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

32. ¿Qué modo de funcionamiento permite a la interfaz de periféricos programable 8255 utilizar un bus bidireccional?
- a) 2
 - b) 0
 - c) 3
 - d) 1
33. Se ha declarado en un programa C la variable int val[5]={1,5,2,1,3}; ¿Cuál de las siguientes afirmaciones es cierta?
- a) val[1] == 1
 - b) &val[3] == (char *)val + 12
 - c) sizeof(val) == 5
 - d) &val[2] es de tipo int * y vale lo mismo que val + 8
34. Un computador usa el formato vertical de codificación de instrucciones para parte de las señales de control y el formato horizontal para k señales de control. El formato vertical posee n campos codificados de m bits cada uno. ¿Cuál es el máximo número de señales de control que pueden usarse en este computador?
- a) $k + n \cdot 2^m$
 - b) $k + n^m$
 - c) $k + n \cdot (2^m - 1)$
 - d) Ninguno de los anteriores
35. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de popcount5:

```
int popcount5(unsigned* array, size_t len)
{
    size_t i,j;
    int x;
    int val, result=0;

    for (i=0; i<len; i++)
    {
        x = array[i];
        val = 0;
        for (j=0; j<=8; j++)
        {
            val += x & 0x01010101;
            x >>= 1;
        }
        val += (val >> 16);
        val += (val >> 8);
    }
}
```

```

        result+= val & 0xFF;
    }
    return result;
}

```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en las condiciones del for interno y el tipo de x.

Esta función `popcount5` fallaría:

- a) con array={1, 16, 256, 4096}
 - b) con array={0x80000000, 0x00400000, 0x00000200, 0x00000001}
 - c) con ambos ejemplos
 - d) con ninguno de los dos ejemplos
36. Las instrucciones JB y JNAE provocan un salto si...
- a) CF == 1
 - b) ZF == 0
 - c) ZF != SF
 - d) SF == 1
37. ¿Qué técnica de E/S se dice controlada por hardware?
- a) E/S programada
 - b) E/S controlada por interrupciones
 - c) Acceso directo a memoria
 - d) Ninguna de las anteriores
38. ¿Cuál de las siguientes afirmaciones sobre el benchmark SPEC CPU es falsa?
- a) La última versión es SPEC CPU2017
 - b) Se cronometran unos 10 tests de enteros y unos 14 tests de punto flotante
 - c) Se usa como referencia un computador UltraSPARC-IV+ 2100MHz, y para cada test se calcula una velocidad cociente entre el tiempo de ejecución en el computador de referencia y en el computador a testear
 - d) El resultado final es la media aritmética de las (10 ó 14) velocidades, bien sea de enteros ó de punto flotante (SPECspeedINT ó SPECspeedFP)
39. ¿Qué método de control de acceso directo a memoria es preferible por velocidad (más rápida), economía (coste no prohibitivo) y conveniencia de diseño (compatible con memorias y sistemas actuales)?
- a) Memoria multipuerto
 - b) Transferencia de bloques o parada de CPU
 - c) DMA intercalado o transparente
 - d) Robo de ciclo

40. Un computador tiene una memoria de control de 640 palabras de 70 bits, de las que 280 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación en lugar de micropogramación?
- a) 19440
 - b) 42280
 - c) 9840
 - d) ninguno de los anteriores resultados es exacto.
41. El sufijo q de la instrucción movq significa:
- a) Que la instrucción realiza un movimiento rápido, trabajando con la mitad menos significativa de los operandos (move quick).
 - b) Que la instrucción realiza un movimiento rápido, trabajando con la mitad más significativa de los operandos (move quick).
 - c) Que la instrucción trabaja con operandos de 64 bits (quad word).
 - d) Que la instrucción trabaja con operandos de 128 bits (quad word).
42. Un controlador de DMA suele ser programado con la siguiente información relativa a una operación de E/S:
- a) tipo de operación, tamaño de bloque a transferir, dirección final de memoria
 - b) tipo de operación, tamaño de bloque a transferir, dirección inicial de memoria
 - c) tipo de operación, dirección inicial de memoria, dirección final de memoria
 - d) Ninguna de las anteriores respuestas es cierta
43. ¿En qué tipos de técnicas de E/S la transferencia de información está bajo el control directo de la CPU?
- a) E/S programada y E/S controlada por interrupciones
 - b) E/S programada y acceso directo a memoria
 - c) E/S controlada por interrupciones y acceso directo a memoria
 - d) Ninguna de las anteriores respuestas es correcta
44. ¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria cache?
- a) Primera
 - b) Segunda
 - c) Tercera
 - d) Cuarta

45. Suponga la siguiente sentencia asm en un programa:

```
asm(" add (%[a],%[i],4),%[r]"  
    :[r] "+r" (result)  
    :[i] "r" (i),  
    [a] "r" (array)  
);
```

¿Cuál de las siguientes afirmaciones es correcta?

- a) a es una posición de memoria de entrada
 - b) i es un registro de entrada
 - c) el código de retorno de la función asm se fuerza a que esté en la variable result
 - d) r es una posición de memoria de entrada/salida
46. Si AX = 0xFA50 y ejecutamos XOR \$0xFF, AX
- a) Se realiza el complemento a 1 de AH.
 - b) Se realiza el complemento a 1 de AL.
 - c) El registro AL se pone a 0.
 - d) El registro AH se pone a 0
47. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros en doble precisión. El programa esqueleto ofrecido (suma.s) no es válido, en concreto...(marcar la opción FALSA)
- a) no está preparado para sumar más de 9 elementos
 - b) no hace extensión con ceros de los elementos
 - c) no hace extensión de signo de los elementos
 - d) no consulta ni el flag de acarreo CF ni el de overflow OF
48. En 80x86, los parámetros a las subrutinas se pueden pasar:
- a) a través de variables globales
 - b) a través de los registros
 - c) a través de la pila
 - d) todas las anteriores son ciertas
49. En la convención SysV AMD64 estándar para arquitecturas x86 de 64 bits, el resultado de una función se devuelve usualmente en el registro:
- a) RBX
 - b) RBP
 - c) RAX
 - d) RSI

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

50. ¿Qué hace gcc -O?
- a) Compilar con optimización suave
 - b) Compilar .c→.o (fuente C a objeto)
 - c) Compilar .s→.o (fuente ASM a objeto)
 - d) Ambas (b) y (c), según la extensión de los ficheros que se usen como argumentos
51. Cuando se produce una interrupción hardware...
- a) Se guarda el estado y se ejecuta la rutina de interrupción asociada
 - b) Se salta a la dirección de memoria indicada en la instrucción actual
 - c) Se aborta la ejecución del programa actual generando un fallo de segmentación
 - d) Se salta al principio del programa actual
52. En un programa en ensamblador queremos crear espacio para una variable entera var inicializada a 1. La línea que hemos de escribir en la sección de datos es:
- a) .int var 1
 - b) var: .int 1
 - c) .int: var 1
 - d) int var 1
53. ¿Cuál de las siguientes afirmaciones es falsa?
- a) Las subrutinas necesitan ser capaces de reservar espacio en memoria para las variables locales sin sobrescribir ningún dato usado por el programa que hace la llamada
 - b) Las subrutinas necesitan recibir parámetros desde el programa que hace la llamada que indiquen qué registros pueden alterar y cuáles no
 - c) Los programas necesitan una forma de pasar parámetros a las subrutinas y de recibir las salidas de vuelta
 - d) Las subrutinas necesitan algún modo de saber desde dónde han sido llamadas para poder volver al programa que realizó la llamada cuando se completa la subrutina
54. Alguna de las siguientes NO es una ventaja de la E/S independiente (separada, aislada)
- a) Protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
 - b) Diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
 - c) Mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
 - d) Decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)

55. ¿Cuál de las siguientes afirmaciones acerca del daisy-chain es cierta?
- a) Todos los componentes se conectan directamente y con igual prioridad al procesador o gestor de interrupciones
 - b) Es incompatible con la técnica de sondeo o polling
 - c) Los componentes se comportan de forma cooperativa: sólo al de mayor prioridad se le concede la interrupción o se apodera del bus de comunicaciones
 - d) Los componentes están conectados todos con todos y un gestor centralizado decide la prioridad
56. ¿Cuáles de las siguientes señales son entradas a la unidad de control?
- a) El contenido del contador de programa
 - b) Las señales de habilitación de buffers triestado entre registros y buses
 - c) El contenido del registro de instrucción
 - d) Las señales de control de la ALU
57. La predicción de saltos está relacionada con...
- a) Los riesgos de (dependencia de) datos (intenta que el dato esté disponible anticipadamente)
 - b) Los riesgos de transferencia (intenta agrupar las posibles transferencias de un conjunto de instrucciones)
 - c) Los riesgos estructurales (intenta evitar el efecto de un fallo de cache)
 - d) Los riesgos de control (intenta determinar de antemano el flujo de control)
58. En el programa "size" de la práctica de la cache, si el primer escalón pasa de tiempo = 1 para todos los tamaños de vector menores o iguales que 32 KB a tiempo = 3 para los tamaños 64 KB y 128 KB, podemos asegurar que:

(Nota: en las opciones de respuesta, "rapidez" se refiere a Ti, el tiempo de acceso efectivo al nivel i, no a ti, el tiempo de acceso propio al nivel i. Si dice "La cache L1 es 3x más rápida que L2", se refiere a que $T1=T2/3$, no a que $t1=t2/3$. Ver T6 tr.26)

-
- a) la cache L2 es como mucho el doble de rápida que la memoria principal
 - b) la cache L1 es al menos tres veces más rápida que la cache L2
 - c) la cache L2 es al menos el doble de rápida que la memoria principal
 - d) la cache L1 es como mucho tres veces más rápida que la cache L2

59. Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos y un computador Y tarda 14 segundos en ejecutar ese mismo programa, ¿cuántas veces es más rápido el computador Y que el X?
- a) 1,538
 - b) 0,538
 - c) 1,857
 - d) 12
60. Para direccionar una memoria de 16K x 16 necesitamos un bus de direcciones de:
- a) 14 bits
 - b) 16 bits
 - c) 4 bits
 - d) Otro valor
61. Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "big-endian", quedará almacenada a partir de la posición 0x1000 como:
- a) M[0x1000]=0xA6 y M[0x1001]=0x91
 - b) M[0x1000]=0x65 y M[0x1001]=0x89
 - c) M[0x1000]=0x89 y M[0x1001]=0x65
 - d) M[0x1000]=0x91 y M[0x1001]=0xA6
62. ¿Cuál de las siguientes afirmaciones sobre memorias cache es cierta? Recordar que llamamos "latencia" al tiempo transcurrido desde que se envía una dirección a la cache hasta que se obtiene el dato (suponiendo que se trata de un acierto), y depende por tanto del tiempo que emplee la circuitería hardware en obtenerlo. Suponer similares condiciones para ambas caches (tamaño de bloque, tamaño total, tecnología, frecuencia...)
- a) Las caches totalmente asociativas ofrecen mejor latencia, mientras que las de correspondencia directa tienen tasas de fallo más bajas
 - b) Las caches totalmente asociativas ofrecen tasas de fallo más bajas, mientras que las de correspondencia directa tienen mejor latencia
 - c) Ambas tienen en general similar latencia y tasa de fallos
 - d) Las caches de correspondencia directa tienen mejor latencia y tasa de fallos
63. En esta asignatura... (marcar la opción FALSA)
- a) se podrían acumular NC=2p y NL=2p respondiendo tests
 - b) Hay que obtener $NTeo \geq 2.5p$ y $NPr \geq 2.5p$, para poder sumar teoría con prácticas y poder aprobar (obtener nota final no suspenso)
 - c) se pueden entregar ejercicios y prácticas aunque cuentan muy poca o ninguna nota
 - d) se podría obtener 10p en el examen final sin asistir un solo día ni a clase ni a prácticas

64. Respecto al sistema de Entrada / Salida, ¿cuál de las siguientes afirmaciones es incorrecta?
- Un protocolo sirve para “ponerse de acuerdo” en cosas como velocidad, paridad, nº de bits, etc.
 - La mayoría de los periféricos trabajan a velocidad muy superior a la CPU; por eso es necesario sincronizar.
 - Un controlador se encarga de la comunicación con la CPU.
 - La CPU se comunica con el periférico por medio del controlador y de software de E/S.
65. ¿Es cierto que la consulta de estado permite averiguar el estado de un periférico y alterar la prioridad de los periféricos?
- Sí, permite averiguar el estado de un periférico y también alterar la prioridad de los periféricos
 - No, porque no permite ni averiguar el estado de un periférico ni alterar la prioridad de los periféricos
 - No, porque no permite averiguar el estado de un periférico
 - No, porque no permite alterar la prioridad de los periféricos
66. El primer nivel de una jerarquía de memoria tiene una tasa de aciertos del 80% y las peticiones de memoria tardan 15 ns en completarse si dicha posición se encuentra en ese nivel y 100 ns si no es así. ¿Cuál es el tiempo medio de acceso de la jerarquía?
- 23 ns
 - 32 ns
 - 57.5 ns
 - 92 ns
67. Cuando dos o más instrucciones necesitan un recurso hardware en el mismo ciclo, se trata de un riesgo:
- estructural
 - de control
 - de salto
 - por dependencia de datos
68. En el arbitraje de un bus...
- los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
 - si hay un único dispositivo pasivo, siempre funciona como esclavo
 - si hay varios dispositivos activos, siempre funcionan como maestros
 - todas las respuestas anteriores son ciertas

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

69. Si un procesador no segmentado necesita 5 ns para leer una instrucción de memoria, 2 ns para decodificar la instrucción, 3 ns para leer del banco de registros, 3 ns para realizar el cálculo requerido por la instrucción, y 2 ns para escribir el resultado en el banco de registros, ¿cuál es la frecuencia de reloj máxima del procesador?
- a) 66,67 MHz
 - b) 500 MHz
 - c) 200 MHz
 - d) 40 MHz
70. ¿Cuál es el resultado de evaluar la expresión $0b1110 \wedge 0b1010$ en lenguaje C?
(0b indica que el número está expresado en binario)
- a) 0b1010
 - b) 0b1111
 - c) 0b0100
 - d) 0b0110
71. De los siguientes grupos de técnicas de E/S, ¿cuáles están controlados por programa?
- a) E/S programada y E/S mediante interrupciones.
 - b) E/S programada y E/S mediante acceso directo a memoria.
 - c) E/S mediante interrupciones y E/S mediante acceso directo a memoria.
 - d) Todas las respuestas anteriores son ciertas.
72. ¿Cuál de las siguientes listas menciona registros x86-64 del mismo tipo respecto a convenio de uso? (salva-invocante, invocado, etc)
- a) RBX, RSI, RDI
 - b) RAX, RBX, RCX, RDX
 - c) CL, DX, R8d, R9
 - d) RSP, RBP
73. Se ha declarado en un programa C la variable `int val[5]={1,5,2,1,3}`. ¿Cuál de las tres primeras opciones (a, b, o c) es FALSA?
- a) `val[1] == 1`
 - b) `&val[3] == val+3`
 - c) `sizeof(val) == 20`
 - d) No se puede marcar ninguna de ellas, todas (a, b y c) son ciertas

74. ¿Cuál de las siguientes es una idea fundamental de la jerarquía de memoria?
- a) Que dispositivos más pequeños y rápidos sirvan de cache para dispositivos más grandes y lentos
 - b) Que dispositivos más grandes y lentos sirvan de cache para dispositivos más pequeños y rápidos
 - c) Crear una gran cantidad de almacenamiento que sea caro y rápido
 - d) Crear una pequeña cantidad de almacenamiento que sea caro y lento
75. ¿Cuál de las siguientes afirmaciones es cierta?
- a) La E/S independiente facilita la protección
 - b) La E/S en memoria es mucho más rápida que la E/S independiente
 - c) La E/S en memoria emplea la patilla IO/M#
 - d) En E/S independiente, las instrucciones de acceso a memoria se emplean tanto para memoria como para E/S
76. Respecto a salvaguardar los registros de la CPU al inicio de una rutina de servicio de interrupción (ISR)
- a) se deben guardar los registros salva-invocado (p.ej. EBX, ESI, EDI en el caso de una CPU IA32), los registros salva-invocante ya los guarda el programa interrumpido
 - b) no es necesario salvar ninguno más, si el contador de programa y los flags de estado ya los salva la propia CPU como parte del mecanismo de interrupción
 - c) se deben guardar los registros que se modifiquen en la propia ISR. Eso es posible hacerlo porque el propio programador de la ISR conoce qué registros va a modificar
 - d) se deben guardar todos los registros, para restaurarlos a la salida y así garantizar que el programa interrumpido no sufre ninguna modificación (salvo el inevitable retraso temporal) debido a la interrupción
77. Según la clasificación m/n, las máquinas de acumulador son de tipo
- a) 0/0
 - b) 2/2 ó 2/3
 - c) 1/1
 - d) ½

78. Sólo una de las siguientes afirmaciones sobre memorias ROM es correcta. ¿Cuál?
- a) Para fabricar una ROM se deben conocer los datos que se desea que almacene
 - b) Una PROM (Programmable ROM) se puede grabar usando un dispositivo programador que selectivamente funde contactos aplicándoles altas temperaturas mediante diminutas cabezas soldadoras (“equipo de puntas”)
 - c) Una EEPROM (Erasable EPROM) se puede grabar (eléctricamente), y borrar (usando rayos ultravioleta)
 - d) Una EPROM (Electrically Progr. ROM) se puede grabar eléctricamente, sin fundir contactos, pero no se puede borrar
79. Ventajas de la E/S independiente (señalar la opción incorrecta)
- a) protección de E/S más fácil (E/S mapeada añade dificultad a la protección de E/S)
 - b) decodificación de memoria más elegante, limpia, sencilla (E/S mapeada añade complejidad a la decodificación)
 - c) mayor aprovechamiento del espacio de memoria (E/S mapeada resta espacio a la memoria)
 - d) diseño del procesador más sencillo (E/S mapeada añade complejidad al diseño)
80. En una jerarquía de memoria, a medida que nos alejamos del procesador:
- a) el coste por byte aumenta
 - b) el tamaño de la unidad de transferencia entre dos niveles aumenta
 - c) el tiempo de transferencia disminuye
 - d) el tamaño de la memoria disminuye
81. ¿Cuál es el tamaño de la marca de cache si el bus de direcciones es de 48 bits (256 TB de memoria principal) y hay 8MB de cache L3, con un tamaño de línea de 64 B y correspondencia asociativa por conjuntos con 16 vías?
- a) 29 bits
 - b) 48 bits
 - c) 13 bits
 - d) 6 bits

82. Un procesador con una unidad de control microprogramada tiene una memoria de control de 340 palabras de 16 bits, de las que 180 son diferentes. ¿Qué ahorro en número de bits obtendríamos si usáramos nanoprogramación?

- a) 19440 bits
- b) 5260 bits
- c) No se produce ahorro
- d) 2560 bits

83. En la secuencia de programa siguiente:

400544: e8 07 00 00 00 callq 400550 <mult2>
400549: 48 89 03 mov %rax,(%rbx)

¿cuál es el valor que introduce en la pila la instrucción call?

- a) 0x400550
- b) 0x400549
- c) 0x40054b
- d) 0x400544

84. En una arquitectura de registros de propósito general (a nivel de lenguaje máquina):

- a) operar usando registros es más rápido.
- b) la generación de código resulta más simple que en arquitecturas de pila o acumulador.
- c) se evita el cuello de botella (por ejemplo, pila, o acumulador) que otras arquitecturas presentan al evaluar expresiones aritméticas complejas
- d) todas las respuestas anteriores son ciertas.

85. ¿Cuál de las siguientes afirmaciones es cierta?

- a) La memoria SRAM es más lenta que la DRAM
- b) La lectura en la memoria SRAM es destructiva
- c) La memoria DRAM es más cara que la SRAM
- d) Ninguna de las anteriores

86. ¿Cuál de los siguientes microprocesadores NO es de 64 bits?

- a) Core i3
- b) Core i7
- c) AVR ATMega
- d) Itanium

87. Suponga la expresión: unsigned int val = x & 0x01010101; donde x es un unsigned int. ¿Cuál de x da como resultado val = 0x01010101?

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

- a) 0x10101010
b) Oaaaaaaaaa
c) Oxeeeeeeee
d) Ox13579bdf
88. ¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?
a) registro-registro
b) registro-memoria
c) memoria-memoria
d) ninguna de las anteriores es cierta
89. En los modos de direccionamiento x86-64 del tipo Desplazamiento(Base,Indice,Factor Escala), puede usarse como
a) base, cualquiera de los 8 registros enteros salvo %rsp
b) índice, también cualquiera salvo %rbp
c) factor de escala, cualquier constante de 1, 2, 4 u 8 bytes
d) desplazamiento, cualquier constante de 1, 2 o 4 bytes (incluso el nombre de una variable, por su dirección)
90. La práctica "popcount" debía calcular la suma de bits (peso Hamming) de los elementos de un array. Un estudiante entrega la siguiente versión de `popcount6`:

```
int popcount6(unsigned* array, size_t len)
{
    size_t i;
    unsigned x;
    int result=0;

    const unsigned m1 = 0x55555553;
    const unsigned m2 = 0x33333333;
    const unsigned m4 = 0x0f0f0f0f;
    const unsigned m8 = 0x00ff00ff;
    const unsigned m16 = 0x0000ffff;

    for (i=0; i<len; i++)
    {
        x = array[i];

        x = (x & m1 ) + ((x >> 1) & m1 );
        x = (x & m2 ) + ((x >> 2) & m2 );
        x = (x & m4 ) + ((x >> 4) & m4 );
        x = (x & m8 ) + ((x >> 8) & m8 );
        x = (x & m16) + ((x >> 16) & m16);
```

```

        result+= x;
    }
    return result;
}

```

Esta función sólo se diferencia de la versión "oficial" recomendada en clase en algún dígito de alguna máscara.

Esta función `popcount6` fallaría:

- a) con array={0xA0000000, 0x00B00000, 0x00000C00, 0x0000000D}
- b) con array={0x10000000, 0x00200000, 0x00000400, 0x00000008}**
- c) con ambos ejemplos
- d) con ninguno de los dos ejemplos

91. ¿Cuál de los siguientes modos de direccionamiento es *menos* preferible para un procesador con segmentación de cauce?

- a) Indirecto a través de memoria**
- b) Indexado (o relativo a base, o base+índice)
- c) Indirecto a través de registro
- d) Registro

92. El lenguaje máquina...

- a) es un conjunto de nombres simbólicos o nemotécnicos.
- b) facilita la portabilidad de los programas.
- c) es el mismo para todos los computadores.
- d) Ninguna de las respuestas anteriores es correcta.**

93. En una CPU de 32 bits con memoria de bytes, el problema es que...

- a) Hay que usar 4 instrucciones de lectura (o escritura) para leer (o escribir) un registro completo
- b) No hay problema, cuando se salva un registro a memoria se escribe en la posición deseada
- c) Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU**
- d) No tiene sentido, un registro no cabría en memoria

94. En la captación de un operando que reside en memoria:

- a) en MBR indicamos la dirección donde está y en MAR lo recogemos
- b) en MAR indicamos la dirección donde está y en IR lo recogemos

- c) en MBR indicamos la dirección donde está y en IR lo recogemos
d) en MAR indicamos la dirección donde está y en MBR lo recogemos
95. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros CON signo produciendo un resultado .quad (doble precisión). El programa esqueleto ofrecido (suma.s) no es válido. Se podría comprobar imprimiendo el resultado SIN signo, y usando como contraejemplo las siguientes listas:
A) 1 y -1 (el usuario piensa que sus datos tienen signo)
B) 0x4000 0000 y 0x4000 0000
C) ninguna de las dos
D) ambas
96. La instrucción movzbl %al, %eax
a) Copia en %eax el valor del indicador de cero
b) Copia en %eax el valor de %al si el indicador de cero está activado
c) Pone a 0 el registro %eax
d) Copia en %eax el valor sin signo almacenado en %al, rellenando con ceros
97. ¿Qué modificador (switch) de ld hace falta para enlazar una aplicación de 32bits en un sistema de 64bits en el que se ha instalado también el compilador de 32bits?
a) No hace falta modificador, ld lo deduce del tipo de objeto a enlazar
b) -32
c) -m elf_i386
d) -m32
98. ¿Cuál de las siguientes secuencias de instrucciones calcula a=b-a, suponiendo que %eax es a y %ebx es b?
a) subl %eax, %ebx
b) subl %ebx, %eax
c) notl %eax
d) addl %ebx, %eax
e) Varias o ninguna de las respuestas anteriores son correctas, no se puede marcar una y sólo una
99. ¿Qué salida produce el siguiente código? Asumir representación de datos de arquitectura x86-64.

```
unsigned int x = 0xDEADBEEF;
```

```

unsigned short y = 0xFFFF;
signed int z = -1;
if (x > (signed short) y)
    printf("Hello");
if (x > z)
    printf("World");

```

(Recordar que:

- 1.- las extensiones de tamaño se hacen según tenga o no signo el tipo fuente
- 2.- en comparaciones con un dato unsigned se pasa el otro dato a unsigned)
 - a) No imprime nada
 - b) Imprime "Hello"
 - c) Imprime "HelloWorld"
 - d) Imprime "World"

100. ¿Qué técnica de E/S consume menos tiempo del procesador?

- a) E/S programada
- b) E/S mediante interrupciones
- c) E/S mediante DMA
- d) Todas pueden consumir el mismo número de recursos en función de la velocidad del dispositivo de E/S al que se acceda

101. ¿Cuál de las siguientes características es posterior a la segunda generación de computadores?

- a) RISC.
- b) Transistor.
- c) Lenguaje ensamblador.
- d) Memoria de núcleos de ferrita.

102. En el siguiente código, ¿qué reordenamiento de los bucles muestra mejor localidad?

```

// X, Y, Z ctes #define previo
int a[X][Y][Z]

```

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

```
int i, j, k, sum = 0;  
for (i = 0; i < Y; i++)  
    for (j = 0; j < Z; j++)  
        for (k = 0; k < X; k++)  
            sum += a[k][i][j];
```

- a) k externo, i central, j interno
- b) j externo, k central, i interno
- c) i externo, j central, k interno (el orden en que están ahora)
- d) El orden de los bucles no afecta a la localidad

103. Sabiendo que la instrucción de llamada mostrada abajo codifica la dirección de la subrutina con direccionamiento relativo a contador de programa (de 32 bits con signo), indicar el valor de los bytes tachados en el siguiente desensamblado.

```
400544: e8 XX XX XX XX callq 400550 <f>  
400549: 48 89 03    mov %rax,(%rbx)
```

- a) 00 40 05 50
- b) 00 00 00 01
- c) 50 05 40 00
- d) 07 00 00 00

104. Una memoria que está organizada en palabras de 8 bits tiene una capacidad de 32 Kbits. ¿Cuántas líneas de dirección tiene dicha memoria?

- a) 32
- b) 8
- c) 4
- d) 12

105. ¿Qué arquitectura es típica en procesadores RISC?

- a) registro-registro
- b) registro-memoria
- c) memoria-registro
- d) memoria-memoria

106. Considerar los siguientes dos bloques de código almacenados en dos ficheros distintos:

```
/* main.c */  
int i = 0;  
int main() {
```

```

func();
return 0;
}

/* func.c */
int i = 1;
void func() {
    printf("%d", i);
}

```

¿Qué sucederá cuando se compile, enlace y ejecute este código?

- a) Escribe "1"
- b) Error al compilar o enlazar, no se obtiene ejecutable**
- c) Escribe "0"
- d) A veces escribe "0" y a veces "1"

107. ¿Qué hace gcc -S?

- a) Compilar .s → .o (fuente ASM a objeto)**
- b) Compilar .c → .s (fuente C a fuente ASM)**
- c) Compilar optimizando tamaño (size), no tiempo
- d) Compilar borrando del ejecutable la tabla de símbolos (strip)

108. En un sistema con memoria de bytes y líneas de cache de 64 bytes, ¿dónde empieza el bloque de memoria que contiene la posición 0xBEE3DE72?

- a) 0xBEE3DE6E
- b) 0xBEE3DE70
- c) 0xBEE3DE40**
- d) 0x0EE3DE72

109. En un sistema con memoria de bytes y líneas de cache de 32 bytes, ¿dónde empieza el bloque de memoria que contiene la posición 0xAC72?

- a) 0xAC60**
- b) 0xAC6E
- c) 0xAC70
- d) 0xAC72

110. Un sistema de memoria tiene un tiempo medio de acceso de 10 ns por operación de lectura o escritura y un ancho de datos de 32 bits. ¿Cuál es el ancho de banda del sistema de memoria?

- a) 2,5 ns
- b) 32 Mbytes por segundo

- c) 400 millones de bytes por segundo
- d) 32 bits
111. ¿Cuántos controladores de interrupciones 8259 hacen falta como mínimo para manejar 17 líneas de interrupción?
- a) 3
- b) 4
- c) 5
- d) 2
112. ¿De qué tipo son los procesadores Intel que usamos en los laboratorios?
- a) el concepto de endian no es aplicable a estas máquinas, ya que un registro del procesador no cabe en una posición de memoria
- b) puede ajustarse mediante un bit de control en el registro CRO que funcionen como little- o big-endian
- c) big-endian
- d) little-endian
113. En el programa "size" de la práctica de la cache, si el primer escalón pasa de tiempo=2 para un tamaño de vector menor que 32KB a tiempo=8 para un tamaño mayor que 32KB, podemos asegurar que:
-
- (Nota: en las opciones de respuesta, "rapidez" se refiere a T_i , el tiempo de acceso efectivo al nivel i , no a t_i , el tiempo de acceso propio al nivel i . Si dice "La cache L1 es 4x más rápida que L2", se refiere a que $T_1=T_2/4$, no a que $t_1=t_2/4$. Ver T6 tr.26)
-
- a) La cache L1 es seis veces más rápida que la cache L2
- b) La cache L1 es cuatro veces más rápida que la cache L2
- c) La cache L1 es como mucho cuatro veces más rápida que la cache L2
- d) La cache L1 es al menos cuatro veces más de rápida que la cache L2
114. Dentro de una función declarada como void swap(long *xp, long *yp), que intercambia los valores de los dos enteros (de tamaño long) cuyas direcciones de memoria (punteros) son pasadas como parámetros a la función, la instrucción movq (%rsi),%rdx copia en %rdx...
- A) el valor del entero apuntado por el puntero pasado como segundo parámetro

- B) el valor del entero apuntado por el puntero pasado como primer parámetro
- C) el valor del puntero pasado como segundo parámetro
- D) el valor del puntero pasado como primer parámetro
115. Respecto a los registros enteros en arquitectura IA32
- a) Hay 8 de cada tamaño (32, 16, 8 bits), aunque no todos los registros tienen versión en 8 y 16 bits
 - b) No hay distintos tamaños, son sólo registros de 32 bits, como corresponde a dicha arquitectura
 - c) Hay 8, y en cada uno puede accederse a todos los 32 bits (p.ej. EAX), a los 16 bits menos significativos (p.ej. AX) ó a los 8 LSBs (p.ej. AL)
 - d) Son de 32bits en las aplicaciones de 32bit, y de 64bits en las aplicaciones de 64bit

116. Tras ejecutar las tres instrucciones que se muestran desensambladas a continuación, el registro RAX toma el valor

08048074 <_start>:

```
8048074: be 74 80 04 08 mov $_start, %rsi
8048079: 46           inc  %rsi
804807a: 8b 06       mov (%rsi), %rax
```

- a) 0x0804807a
- b) 0x08048075
- c) 0x08048079
- d) 0x08048074

117. Usando el repertorio x86-64, para intercambiar el valor de 2 registros se pueden usar...

- a) 3 mov, no menos (se le llama "intercambio circular")
- b) una instrucción mov y una instrucción lea
- c) dos instrucciones mov
- d) 4 mov, no menos (debido a la arquitectura R/M)

118. El fragmento de código:

```
poll: in a, 0x20
      cmp a, $0
      jnz poll
      load a, 0x11
```

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

out 0x21, a

corresponde a:

- a) E/S programada con consulta de estado
- b) E/S programada sin consulta de estado
- c) E/S por DMA
- d) E/S por interrupciones

119. ¿Cuál de los siguientes es el ejemplo más acertado de localidad espacial?

- a) Iterar repetidamente el cuerpo de un bucle
- b) Reservar dinámicamente (malloc) espacio para una estructura o union
- c) Referenciar elementos de un array sucesivamente
- d) Referenciar continuamente la misma variable local

120. Las interrupciones generadas por el teclado interrumpirán al procesador:

- a) sólo si el procesador tiene activado el indicador de habilitación de interrupciones
- b) siempre que el usuario pulse una tecla en el teclado
- c) sólo si el procesador está chequeando el estado del teclado
- d) sólo cuando el procesador no esté realizando un trabajo útil

121. Alguna de las siguientes líneas de código sirve para definir una variable entera llamada tam en GNU/as Linux x86. ¿Cuál?

- a) tam: .int .-msg
- b) _int tam = 0
- c) int tam;
- d) var tam : integer;

122. Supongamos dos procesadores con bus de direcciones con idéntico número de líneas. Si uno de ellos emplea E/S mapeada en memoria y el otro E/S independiente, ¿cuál podrá acceder a una mayor cantidad de memoria?

- a) Depende del tamaño del bus de direcciones
- b) Ambos podrán acceder a la misma cantidad de memoria
- c) El que tiene E/S independiente
- d) El que tiene E/S mapeada en memoria

123. El ancho de palabra de una memoria corresponde a:

- a) El número que identifica únicamente cada posición de la memoria.
- b) La cantidad de bits que caben en una sola posición
- c) El número de posiciones que la componen.
- d) La longitud del registro de direcciones de la memoria.

124. Alguno de los siguientes NO es un motivo de que no se alcance la ganancia ideal en un cauce segmentado

- a) La duración del ciclo de reloj impuesta por la etapa más lenta
- b) El propio coste de la segmentación (carga de los registros de acople, etc...)
- c) Los riesgos (hazards)
- d) La emisión múltiple (y posiblemente desordenada) de instrucciones

125. Si se dice que en un sistema computador cada dirección especifica uno o dos puertos de E/S, se refiere a que:

- a) La misma dirección puede usarse para transferir un byte o una palabra de mayor tamaño (ese byte y el siguiente)
- b) La misma dirección (por ejemplo 0x0210) puede ser una posición de memoria o un puerto de E/S, según IO/M#
- c) La pregunta es capciosa, una dirección puede especificar un puerto, no dos
- d) Un puerto será de sólo lectura, otro de sólo escritura, y ambos se decodifican en la misma dirección

126. Si la dirección del primer elemento de un vector de enteros z está almacenada en el registro %rdi y el índice i está almacenado en el registro %rsi, la instrucción máquina que realiza la operación $z[i]++$ es:

- a) addl \$4, (%rsi,%rdi)
- b) addl \$1, (%rsi,%rdi,4)
- c) addl \$1, (%rdi,%rsi)
- d) addl \$1, (%rdi,%rsi,4)

127. Sea un formato de microinstrucción que incluye dos campos independientes de 8 bits cada uno. Si se rediseña de modo que se solapen los dos campos, ¿cuántos bits se ahorrán en cada microinstrucción?

- a) 9
- b) 7
- c) 1
- d) 8

128. Utilizando la sentencia `asm()`, las denominadas restricciones que se indican al final de dicha sentencia, involucran a:

- a) Solamente las entradas
- b) Solamente las salidas
- c) Solamente los sobrescritos
- d) Ninguna de las anteriores es cierta

129. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):

- a) movb %sil, (%rax)
- b) pushq \$0xFF
- c) movsbw (%rax), %dx
- d) movzlq %edx, %rax

130. La instrucción test es...

- a) Lo mismo que sub, pero no guarda el resultado, sólo ajusta los flags
- b) Lo mismo que and, pero no guarda el resultado, sólo ajusta los flags
- c) Lo mismo que sub
- d) Lo mismo que and

131. La diferencia entre las instrucciones test y cmp consiste en que

- a) test realiza una operación and lógico, mientras que cmp realiza una resta
- b) modifica sólo los flags lógicos (ZF,SF) mientras que cmp modifica los aritmético-lógicos (ZF,SF,CF,OF)
- c) ambas respuestas son correctas
- d) ambas respuestas son incorrectas

132. Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria.

Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de 2^{10} , ¿qué tamaño de memoria admite el computador?

- a) 2^{10} palabras
- b) 2^{12} palabras
- c) 2^{13} palabras
- d) Ninguna de las anteriores

133. De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...

- a) El bucle for
- b) El bucle do-while
- c) La selección switch-case
- d) El bucle while

134. Respecto a registros salva-invocante y salva-invocado en GCC/Linux x86-64, ¿cuál de éstos es de distinto tipo que el resto?

- a) RAX
- b) RBX
- c) RSI
- d) RDI

135. En un procesador de la familia 80x86 las posiciones de memoria que representan una variable int (entero 4B compl.2) contiene los bytes: F0 FF FF FF. ¿Cuánto vale dicha variable?
- a) 4043309055
 - b) 16
 - c) 4294967280
 - d) -16

136. Considerar la declaración C

```
long array[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
```

Suponer que el compilador tiene la dirección de array en el registro %rcx. ¿Cómo se

movería el valor array[3] al registro %rax? Asumir que %rbx es 3.

- a) leaq (%rcx,%rbx,8),%rax
- b) leaq 12(%rcx),%rax
- c) movq 8(%rcx,%rbx,2),%rax
- d) movq (%rcx,%rbx,8),%rax

137. Por x86-64 se entiende la misma arquitectura de repertorio (ISA) que

- a) IA32
- b) AMD64
- c) x86
- d) IA64

138. El direccionamiento relativo a registro base utiliza...

- a) dos registros.
- b) un registro y un desplazamiento.
- c) un registro y un factor de escala
- d) dos desplazamientos contenidos en la propia instrucción.

139. Sobre el direccionamiento relativo a contador de programa:

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos y llamadas a subrutinas reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está ejecutando.
- d) Todas las respuestas son ciertas.

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

140. Un sistema con direcciones de 8bits utiliza una puerta NAND conectada a las líneas A7...A5 para atacar la entrada CS# (activa baja) de un módulo de memoria. En el mapa de memoria las siguientes posiciones corresponderán a dicho módulo
- a) 0x70 a 0x7f y 0xf0 a 0xff
 - b) 0x00 a 0x0f y 0x80 a 0x8f
 - c) **0xe0 a 0xffff**
 - d) 0x00 a 0x1f
141. En EC podemos usar la palabra directo para referirnos a... (señalar la opción incorrecta)
- a) modo de direccionamiento directo
 - b) **dispositivo de almacenamiento secuencial directo (DASD)**
 - c) acceso directo a memoria
 - d) cache con correspondencia directa
142. ¿Cuál de las siguientes secuencias de tipos de memoria está ordenada de menores a mayores prestaciones?
- a) SDRAM, DDR, EDO
 - b) **FPM, EDO, RDRAM**
 - c) EDO, SRAM, FPM
 - d) DDR, SDRAM, FPM
143. ¿Cuántos bits hacen falta como mínimo para implementar tres niveles de inhibición de interrupciones (general, nivel y máscara) en un sistema con cuatro niveles de interrupción?
- a) 6
 - b) 5
 - c) 4
 - d) 7**
144. Si %rdx contiene 0xf000 y %rcx contiene 0x0100, el direccionamiento 0x80(%rcx,%rdx,2) se refiere a la posición
- a) 0xf182
 - b) 0xf280
 - c) 0x1e180**
 - d) Ninguna de las respuestas anteriores es correcta
145. La ganancia en velocidad ideal de un cauce de K etapas de igual duración T ejecutando un programa de N instrucciones es
- a) $S = KN/(K+N-1)$**
 - b) $S = KN/(K-N+1)$
 - c) $S = NT/(N+K-1)T$
 - d) $S = NKT/(N-K+1)T$

146. ¿Por qué se impusieron las arquitecturas de registros de propósito general a las arquitecturas basadas en pila?

- a) Porque no se puede programar una arquitectura de pila en un lenguaje de alto nivel
- b) Porque las basadas en registros permiten reducir el tamaño del programa
- c) Porque la memoria es más cara que los registros
- d) Porque las basadas en registros son capaces de lograr un mejor rendimiento cuando se asignan variables a registros

147. ¿Qué política de colocación en cache necesita más comparadores, la correspondencia asociativa por conjuntos o la correspondencia por sectores?

- a) Depende de si es mayor el número de conjuntos o el número de sectores
- b) Depende de si es mayor el número de bloques por conjunto o el número de sectores
- c) Correspondencia por sectores
- d) Correspondencia asociativa por conjuntos

148. En una cache con 64 bytes de longitud de línea, ¿qué bits de una dirección de memoria de 64 bits se utilizan para determinar a qué byte dentro de la línea se refiere dicha dirección? (Memoria direccionable por bytes)

- a) [11...6]
- b) [5...0]
- c) [8...6]
- d) [5...3]

149. ¿Cuál de las siguientes afirmaciones es cierta?

- a) En el direccionamiento a registro, el objeto direccionado es una constante contenida en la propia instrucción.
- b) Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.
- c) El modo de direccionamiento indirecto a través de registro es el modo más sencillo de direccionamiento a memoria.
- d) En el direccionamiento inmediato, la dirección efectiva se encuentra en un registro de uso general del procesador.

150. ¿Cuál es el número máximo de niveles de interrupción que se pueden manejar utilizando 3 controladores de interrupciones programables 8259?

- a) 18
- b) 20
- c) 22
- d) 24

151. Considere un sistema de memoria para un procesador de 32 bits con caches separadas para código y datos. Suponga que el procesador direcciona la memoria por bytes y realiza accesos a palabras de 32 bits y que el espacio de direcciones es de 2³² bytes. La cache de datos tiene las siguientes características: 64 KB de capacidad, asociativa por conjuntos con 2 vías, y bloques de 2 palabras. ¿Cuántos bits tiene el campo etiqueta de una dirección de memoria?

- A) 17
- B) 13
- C) 15
- D) 11

152. El lenguaje máquina es...

- a) portable entre arquitecturas.
- b) fácilmente legible por el programador.
- c) difícil de codificar manualmente.
- d) una alternativa razonable al uso del lenguaje ensamblador.

153. En una máquina little-endian con memoria de bytes y representación en complemento a dos que permite accesos a memoria de tamaño byte (1B), media palabra (2B) y palabra (4B), si se almacena en la posición 0xBABC una palabra de valor -2, ¿qué se obtendría al consultar la media palabra en la posición 0xBABE?

- A) 0
- B) -2
- C) 1
- D) -1

154. ¿Qué técnica de E/S requiere menos atención por parte del procesador?

- a) Todas requieren la misma atención
- b) E/S mediante interrupciones
- c) E/S mediante acceso directo a memoria
- d) E/S programada

155. ¿Qué instrucciones son típicas del repertorio de un procesador que sólo disponga de E/S mapeada en memoria?

- a) IN, LOAD, OUT
- b) IN, LOAD, MOV
- c) LOAD, MOV, STORE
- d) Ninguno de los anteriores

156. Si el tiempo de acceso a la memoria cache es de 2 ns y el tiempo necesario para tratar un fallo de cache es de 80 ns, ¿cuál es la tasa de aciertos necesaria para que el tiempo medio de acceso al sistema de memoria sea de 10 ns?
- a) 0.9
 - b) 0.95
 - c) 0.75
 - d) 0.8
157. ¿Qué forma tiene la gráfica que se debe obtener con el código size.cc?
- a) Forma de U, con un tramo descendente y otro ascendente.
 - b) Forma de ∩, con un tramo ascendente y otro descendente.
 - c) Una escalera decreciente con varios tramos horizontales.
 - d) Una escalera creciente con varios tramos horizontales.
158. Si queremos almacenar la palabra de 64bits 0x0000001f ffffffe0 en una memoria de bytes según la convención little-endian a partir de la posición 0x0804913c, quedará
- a) 0x00 en 0x0804913c y 0xe0 en 0x08049143
 - b) 0x1f en 0x0804913c y 0xe0 en 0x08049140
 - c) 0xe0 en 0x0804913c y 0x1f en 0x08049140
 - d) Todas las respuestas anteriores son incorrectas
159. Justo antes de que una instrucción máquina escriba un resultado en memoria:
- a) en MBR está el resultado y en MAR la dirección donde se almacenará
 - b) en IR está el resultado y en MAR la dirección donde se almacenará
 - c) en IR está el resultado y en MBR la dirección donde se almacenará
 - d) en MAR está el resultado y en MBR la dirección donde se almacenará
160. ¿Cuál de las siguientes funciones no corresponde a un controlador (interfaz) de E/S?
- a) Almacenamiento de programas
 - b) Almacenamiento temporal de datos
 - c) Comunicación con el microprocesador
 - d) Comunicación con el dispositivo
161. El bus del sistema es
- a) el que conecta las distintas partes del sistema: UC, ALU, E/S, M
 - b) en un sistema con buses separados, el que conecta el sistema E/S con el resto
 - c) el que conecta CPU-M, ya sea un sistema con bus único o con múltiples buses
 - d) en un sistema con bus único, todo el bus salvo la parte relacionada con E/S (SATA, GPU, USB, Ethernet, etc)

MARVEL STUDIOS
Bruja Escarlata
y Visión



Disney+

Serie Original ya disponible
en exclusiva

162. En la práctica "media" se pide como ejercicio previo sumar una lista de N enteros en doble precisión. El programa esqueleto ofrecido (suma.s) no es válido, en concreto...
(marcar la opción FALSA)
- a) no está preparado para sumar más de 9 elementos
 - b) no hace extensión con ceros de los elementos
 - c) no hace extensión de signo de los elementos
 - d) no consulta ni el flag de acarreo CF ni el de overflow OF

1. ¿De qué depende el tamaño del contador de programa?
 a) De la longitud del código de operación de las instrucciones.
 b) Del ancho del bus de datos.
 c) Del número de instrucciones diferentes y de los tipos de direccionamiento posibles.
 d) **Del número de direcciones de memoria.**
2. Para direccionar una memoria de bytes en la que quepan 2G palabras de 32 bits se necesitarán:
 a) 31 bits como mínimo
 b) 32 bits exactamente
 c) **33 bits como mínimo**
 d) 21 bits como máximo
3. ¿De qué depende el tamaño del contador de programa?
 a) de la longitud del código de operación
 b) el tamaño no importa
 c) del ancho del bus de datos
 d) **ninguna de las otras respuestas es cierta**
4. Una computadora puede funcionar prescindiendo de:
 a) del PC
 b) **de un acumulador**
 c) del IR
 d) de los biestables de condición
5. Si usamos una estructura de bus con DMA:
 a) **la CPU puede dejar las transferencias entre MP y periféricos en manos de este controlador (DMA) y seguir ejecutando otras instrucciones.**
 b) al bus del sistema sólo se conecta la CPU y la MP, ya que el DMA se conecta directamente a MP para realizar las transferencias de datos.
 c) la velocidad de este controlador establece la velocidad del bus del sistema.
 d) podemos prescindir de controladores de E/S ya que el controlador de DMA se ocupa de controlar las transferencias hacia/desde los periféricos.
6. Si un computador X ejecuta un programa de 450 millones de instrucciones en 26 segundos y un computador Y tarda 14 segundos en ejecutar ese mismo programa, ¿cuántas veces es más rápido el computador Y que el X?
 a) 12
 b) 0,538
 c) **1,857**
 d) 1,538
7. ¿Cuál de las siguientes afirmaciones es incorrecta?

- a) En lenguaje ensamblador cada instrucción se corresponde con una instrucción máquina.
 - b) Un programa de alto nivel interpretado es más lento que el mismo programa en lenguaje máquina.
 - c) El lenguaje de alto nivel es más portable que el lenguaje máquina.
 - d) **En lenguaje ensamblador las instrucciones se escriben en binario.**
8. ¿Cómo se almacenaría como palabra de 32 bits el número -128 en un sistema que utilice el criterio del extremo menor ("little endian")?
- a) posición 0: FF pos.1:FF pos.2: FF pos.3: 80
 - b) **0:80 1:FF 2:FF 3:FF**
 - c) 0:00 1:01 2: 00 3:80
 - d) Ninguna de las anteriores
9. En las últimas generaciones de computadores la mejora de prestaciones viene dada por:
- a) avances en la estructura y arquitectura del computador.
 - b) **avances en tecnología y avances en la estructura y arquitectura del computador.**
 - c) avances en las tecnologías de fabricación.
 - d) avances en los sistemas operativos y aplicaciones.
10. ¿Qué modelo de programa se ejecuta en las arquitecturas Von Neumann?
- a) Externo.
 - b) Microprogramado.
 - c) Cableado.
 - d) **Almacenado**

Si queremos almacenar la palabra de 16 bits 0x9660 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x1000 como:

- a) M[0x1000]=0x06 y M[0x1001]=0x69
- b) M[0x1000]=0x69 y M[0x1001]=0x06
- c) M[0x1000]=0x96 y M[0x1001]=0x60
- d) M[0x1000]=0x60 y M[0x1001]=0x96

El computador EDVAC, propuesto por John Von Neumann, presentaba dos importantes diferencias respecto al ENIAC:

- a) **Empleaba aritmética binaria y permitía trabajar con un programa almacenado.**
- b) Utilizaba transistores y memoria de semiconductor.
- c) Se programaba enchufando centenares de clavijas y empleaba aritmética octal.
- d) Era electromecánico y de propósito específico.

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) El modo de direccionamiento permite determinar un operando o la ubicación del operando.
- b) Los operandos siempre están almacenados en memoria.
- c) El repertorio de instrucciones es el conjunto de operaciones que es capaz de interpretar la unidad de control.
- d) El repertorio de instrucciones debe ser capaz de realizar una tarea en un tiempo finito.

El registro MBR...

- a) especifica la dirección en memoria de la palabra que va a ser escrita o leída
- b) contiene el código de operación de la instrucción que se está ejecutando
- c) contiene la dirección de la próxima instrucción que va a ser captada de memoria
- d) contiene el valor que va a ser almacenado en la memoria, o bien se usa para recibir un valor procedente de la memoria

Una dirección de memoria se refiere siempre a:

- a) una palabra
- b) 16 bits
- c) un byte
- d) ninguna de las anteriores

En las instrucciones aritméticas con dos operandos de un procesador con arquitectura de pila, los dos operandos...

- a) son dos registros del procesador.
- b) se introducen en la pila tras realizar la operación.
- c) son la cima de la pila y el elemento siguiente de la cima de la pila.
- d) pueden estar en cualquier posición de la pila.

En un sistema con dos buses separados, uno para el subsistema de memoria y otro para la E/S...

- a) el bus que une la memoria y el procesador suele funcionar a la velocidad de la memoria
- b) el bus de E/S funciona a la velocidad del periférico más rápido
- c) ambos buses tienen que tener el mismo ancho de banda
- d) Ninguna de las respuesta anteriores es cierta

La idea de desarrollar máquinas CISC surgió para:

- a) simplificar el diseño hardware de la UC.
- b) tener instrucciones cercanas al lenguaje de alto nivel.
- c) conseguir un conjunto de instrucciones cortas y sencillas de decodificar.
- d) ninguna de las respuestas anteriores es cierta.

Un modo de vídeo de 512 x 256 píxeles y 16 colores por píxel ocupa una memoria de:

- a) 8 KB
- b) 64 KB
- c) 2 MB
- d) 128 KB

Un computador con 8 bits en el bus de direcciones puede direccionar como máximo:

- a) 8192 palabras
- b) 16384 palabras
- c) 256 palabras
- d) 1024 palabras

¿Cuál es el contenido de la pila al terminar de ejecutarse la siguiente secuencia de instrucciones de una arquitectura de pila?:

```
push #4  
push #7  
add  
push #10  
sub
```

- a) 11, 1
- b) 1
- c) 4, 7, 10
- d) 4, 11, 1

Para direccionar una memoria de bytes en la que quepan 1G palabras de 32 bits se necesitarán:

- a) 32 bits
- b) 33 bits
- c) 21 bits
- d) 31 bits

¿En qué generación, dentro de la historia de los computadores digitales, aparece la microprogramación?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

Un computador que utilice el sistema big-endian, almacena el número 0x2143 a partir de la dirección 0 como:

- a) M[0]=0x43 y M[1]=0x21
- b) M[0]=0x21 y M[1]=0x43
- c) M[0]=0x34 y M[1]=0x12

MARVEL STUDIOS
Bruja Escarlata
y Visión



Disney+

Serie Original ya disponible
en exclusiva

- d) $M[0]=0x12$ y $M[1]=0x34$

Indique cuál de las siguientes afirmaciones sobre el ENIAC no es correcta:

- a) Un inconveniente era que tenía que ser programado manualmente mediante commutadores, y conectando y desconectando cables.
- b) La idea del programa almacenado surgió durante el desarrollo del ENIAC, pero no fue implementada en este computador.
- c) Contaba con más de 17000 tubos de vacío.
- d) Como los computadores actuales, era una máquina binaria, es decir, los números estaban representados en forma binaria y los cálculos aritméticos se realizaban también en el sistema binario.

En una estructura de computador de bus único (bus del sistema):

- a) es la estructura más usada en los PC actuales
- b) la UC concede el acceso al bus, por lo que éste funciona a la velocidad de la CPU
- c) sólo una unidad funcional puede tener el control del bus en cada momento
- d) ninguna de las otras respuestas es cierta

El objetivo de un diseño CISC es...

- a) disminuir el tamaño medio de instrucción.
- b) disminuir el número de instrucciones a ejecutar por un programa.
- c) disminuir el número medio de ciclos por instrucción.
- d) disminuir la frecuencia de reloj.

¿Qué parámetro es más importante para comparar la velocidad de dos ordenadores diferentes?

- a) El resultado de la ejecución de un conjunto de programas de prueba.
- b) La arquitectura del procesador.
- c) El número medio de ciclos de reloj por instrucción.
- d) La frecuencia de reloj del procesador.

¿En qué generación, dentro de la historia de los computadores digitales, aparecen los sistemas operativos multiusuario?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

En el arbitraje de un bus...

- a) los dispositivos pasivos pueden requerir el uso del bus para iniciar una transferencia
- b) si hay un único dispositivo pasivo, siempre funciona como esclavo
- c) si hay varios dispositivos activos, siempre funcionan como maestros

- d) todas las respuestas anteriores son ciertas

¿Cuál de las siguientes características es posterior a la segunda generación de computadores?

- a) RISC.
- b) Memoria de núcleos de ferrita.
- c) Transistor.
- d) Lenguaje ensamblador.

¿Cuál de las siguientes no es una característica de los computadores RISC?

- a) La decodificación de las instrucciones debe ser simple: un computador RISC debería emplear un único formato de instrucción
- b) Un computador RISC no debe emplear microprogramación.
- c) Para acelerar el computador RISC se emplean técnicas de pipelining.
- d) Las funciones que realizan los computadores RISC deben ser lo más complejas y potentes que sea posible.

¿En qué generación, dentro de la historia de los computadores digitales, se alcanzaron tiempos de conmutación del orden de nanosegundos?

- a) primera
- b) segunda
- c) tercera
- d) cuarta

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) Todas las instrucciones deben tener código de operación
- b) El formato de una instrucción nos indica el significado de cada bit de la instrucción
- c) Todas las instrucciones deben tener operando fuente y operando destino
- d) No siempre es necesario indicar la dirección de la siguiente instrucción

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "big-endian", quedará almacenada a partir de la posición 0x1000 como:

- a) M[0x1000]=0xA6 y M[0x1001]=0x91
- b) M[0x1000]=0x65 y M[0x1001]=0x89
- c) M[0x1000]=0x91 y M[0x1001]=0xA6
- d) M[0x1000]=0x89 y M[0x1001]=0x65

En el contexto del lenguaje máquina, el acrónimo ISA suele referirse a:

- a) Information Security Architecture
- b) Instruction Set Architecture
- c) Intel Standard Architecture

- d) Industry Standard Architecture

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x8600 como:

- a) $M[0x8600]=0x69$ y $M[0x8601]=0x85$
- b) $M[0x8600]=0x65$ y $M[0x8601]=0x89$
- c) $M[0x8600]=0x89$ y $M[0x8601]=0x65$
- d) $M[0x8600]=0x85$ y $M[0x8601]=0x69$

¿Qué tipo de instrucciones se emplean más en una arquitectura de acumulador?

- a) aritmético-lógicas
- b) **de transferencia de datos con memoria**
- c) de desplazamiento y rotación
- d) de transferencia de datos entre registros

En la arquitectura Von Neumann...

- a) **el programa se encuentra residente en memoria.**
- b) Todas son ciertas.
- c) los bloques principales son la unidad de control, la ALU y la CPU.
- d) los registros se encuentran en la memoria principal.

¿Cuál de las siguientes afirmaciones es falsa?

- a) **la anchura del bus de datos es de 16 bits**
- b) el bus de direcciones es unidireccional
- c) el bus de control transporta señales de estado
- d) el bus de datos es bidireccional

En las arquitecturas RISC hay...

- a) **muchos registros y pocos modos de direccionamiento.**
- b) pocos modos de direccionamiento y muchos formatos de instrucción.
- c) pocos registros y muchos tipos de instrucciones.
- d) pocas instrucciones muy rápidas con muchos modos de direccionamiento.

¿En qué generación, dentro de la historia de los computadores digitales, aparece la memoria virtual?

- a) primera
- b) segunda
- c) **tercera**
- d) cuarta

No en todas las instrucciones máquina hay una fase de

- a) decodificación
- b) ejecución
- c) captación
- d) **captura de operandos**

¿Cuál de las siguientes afirmaciones acerca de la arquitectura Von Neumann es cierta?

- a) La arquitectura Von Neumann es un diseño que sitúa el programa en un almacenamiento distinto al de los datos.
- b) Para ejecutar un programa en una máquina Von Neumann, es necesario volver a cablear o incluso rediseñar la máquina.
- c) **La separación entre almacenamiento y unidad de procesamiento es una de los ideas contempladas en la arquitectura Von Neumann.**
- d) Existe un consenso general en considerar justo el término "arquitectura von Neumann", ya que las ideas de esta arquitectura fueron completamente originales de John Von Neumann y no influenciadas por sus colaboradores o predecesores.

¿Qué dice la ley de Moore?

- a) Todas las respuestas son ciertas.
- b) Que el tamaño de los transistores se duplica cada 18 meses.
- c) **Que el número de transistores de un chip se duplica cada 18 meses.**
- d) Que la memoria de los ordenadores se duplica cada 18 meses.

Una CPU con bus de direcciones de 16 bits y bus de datos de 8 bits tiene un registro de 8 bits conectado al bus de datos y a la unidad de control. Puede tratarse del registro

- a) Contador de programa
- b) **De instrucción**
- c) Puntero de pila
- d) De direcciones

Una máquina superescalar es aquella que:

- a) basa su funcionamiento en la segmentación software como forma de incrementar el paralelismo.
- b) las instrucciones tienen un campo por cada unidad funcional al realizarse varias operaciones por instrucción.
- c) **emite simultáneamente múltiples instrucciones por ciclo de reloj, por ejemplo, una entera y otra de coma flotante.**
- d) ninguna respuesta de las anteriores es correcta.

¿Qué modelo de programa se ejecuta en las arquitecturas de Von Neumann?

MARVEL STUDIOS
**Bruja Escarlata
y Visión**



Disney+

Serie Original ya disponible
en exclusiva

- a) microprogramado
- b) nanoprogramado
- c) **almacenado**
- d) cableado

Un computador con 13 líneas de direcciones utiliza E/S mapeada a memoria. Si se supone que cada uno de los periféricos ocupa 4 direcciones y que el número de periféricos que se planea conectar es de 2^{10} , ¿qué tamaño de memoria admite el computador?

- a) 2^{10} palabras
- b) **2^{12} palabras**
- c) 2^{13} palabras
- d) Ninguna de las anteriores

¿Qué arquitectura se caracteriza por presentar una gran variación en la longitud de las instrucciones?

- a) registro-registro
- b) registro-memoria
- c) **memoria-memoria**
- d) ninguna de las anteriores es cierta

En la captación de un operando que reside en memoria:

- a) en MBR indicamos la dirección donde está y en MAR lo recogemos
- b) en MBR indicamos la dirección donde está y en IR lo recogemos
- c) **en MAR indicamos la dirección donde está y en MBR lo recogemos**
- d) en MAR indicamos la dirección donde está y en IR lo recogemos

En el direccionamiento inmediato, tras captarse completamente la instrucción:

- a) se accede al operando, que está contenido en una posición de memoria principal.
- b) se accede al operando, que se encuentra almacenado en uno de los registros programables.
- c) **se accede al operando, que es una constante contenida en la propia instrucción.**
- d) el código de operación contiene el operando.

El espacio direccionable de memoria de un computador depende del diseño del:

- a) **Bus de direcciones**
- b) Bus de datos
- c) a) y b) son correctas
- d) Ninguna de las anteriores es correcta

La primera generación de computadores se caracteriza por el uso de:

- a) Microprocesadores.
- b) Fibra óptica.
- c) Transistores.
- d) **Tubos de vacío.**

¿En qué registro está contenido el último dato (o instrucción) leído de memoria, o el dato que se va a escribir en memoria?

- a) MAR.
- b) **MBR.**
- c) Acumulador.
- d) PC.

El ancho de palabra de una memoria corresponde a:

- a) La longitud del registro de direcciones de la memoria.
- b) El número de posiciones que la componen.
- c) El número que identifica únicamente cada posición de la memoria.
- d) **La cantidad de bits que caben en una sola posición**

¿Cuál de las siguientes afirmaciones es incorrecta?

- a) **El tamaño de una instrucción en lenguaje máquina siempre ocupa dos bytes en los procesadores RISC.**
- b) En las arquitecturas CISC hay más instrucciones que en las RISC.
- c) Las arquitecturas RISC son del tipo registro-registro.
- d) Las arquitecturas RISC simplifican la decodificación.

¿Qué medida de prestaciones es la más fiable de todas las posibles?

- a) **ninguna de las otras respuestas es correcta**
- b) MIPS equivalentes
- c) MFLOPS
- d) MIPS

¿Cuál es el contenido de una pila al terminar de ejecutarse la siguiente secuencia de operaciones push y pop?:

```
push #1  
push #2  
push #3  
pop a  
push #4  
pop a  
pop a
```

- a) 1 y 2
- b) 10

c) 1

d) 1, 2, 3 y 4

El programador de lenguaje ensamblador necesita conocer:

- a) el diseño RTL del procesador.
- b) todas las respuestas son ciertas.
- c) la arquitectura del ordenador.
- d) la microarquitectura del procesador.

¿En qué generación, dentro de la historia de los computadores digitales, aparecen las memorias de semiconductores?

- a) Segunda generación.
- b) Tercera generación.
- c) Cuarta generación.
- d) Quinta generación.

El direccionamiento directo a memoria utiliza...

- a) Dos desplazamientos contenidos en la misma instrucción.
- b) Un registro y un desplazamiento contenidos en la propia instrucción.
- c) Un desplazamiento.
- d) Un registro.

El ancho de palabra de una memoria corresponde a:

La longitud del registro de datos de la memoria.

Siendo EDX=0xf000 y ECX=0x100 ¿Cuál de las siguientes instrucciones tiene como dirección efectiva 0xf400?

xorl (%edx,%ecx,4), %eax.

Para direccionar una memoria de 1 G palabras de 32 bits cada una, que se direcciona byte a byte, se necesitarán:

32bits.

Un modo de vídeo de 1680 x 1050 píxeles y 4 canales de color por píxel (R, G, B, alpha) y 256 niveles de intensidad por canal (0 a 255), ocupa aproximadamente una memoria de:

- a) 7 MB
- b) 1,7 GB
- c) 54 MB
- d) 1,7 MB

Suponiendo que todos los registros inicialmente contienen el valor 1, ¿cuál es el valor de ECX tras la ejecución de la siguiente secuencia de instrucciones?

```
mov $4, %eax  
mov $3, %ebx  
lea (%eax, %eax), %ecx  
sub %ebx, %ecx  
imul %ecx, %eax
```

- a) 6
- b) 36
- c) 20
- d) 5

En una arquitectura de registros (a nivel de lenguaje máquina):

- a) operar usando registros es más rápido.
- b) la generación de código resulta más simple que en arquitecturas de pila o acumulador.
- c) tiene el problema de que las instrucciones pueden ser largas.
- d) **todas las respuestas anteriores son ciertas.**

En las arquitecturas RISC:

- a) **se usan instrucciones muy simples que se pueden segmentar.**
- b) la programación resulta más simple que en los CISC.
- c) la UC suele ser más compleja que en CISC, al ser cableada.
- d) realmente se usan pocas instrucciones de las disponibles en el conjunto de instrucciones.

En una máquina con 32 registros direccionables e instrucciones de 16 bits es posible codificar:

- a) 64 instrucciones de dos registros y 32 instrucciones de un registro.
- b) 2 instrucción de tres registros y 32 instrucciones de dos registros.
- c) **1 instrucción de tres registros, 31 instrucciones de dos registros, 31 instrucciones de un registro y 32 instrucciones de 0 direcciones.**
- d) 63 instrucciones de dos registros, 32 instrucciones de un registro y 32 instrucciones de 0 direcciones.

En el 8086, la dirección efectiva de la cabecera de pila vendrá dada por:

- a) **SS * 10h + SP**
- b) SS * 10h + BP
- c) SP
- d) BP

Sobre el direccionamiento relativo a contador de programa:

- a) Favorece la implementación de código reubicable.
- b) Su uso en los saltos y llamadas a subrutinas reduce el tamaño de la instrucción.
- c) Es adecuado para alcanzar instrucciones próximas a la que se está



Serie Original ya disponible
en exclusiva



ejecutando.

- d) **Todas las respuestas son ciertas.**

En el RISC-I, una ventana de registros contiene, entre otros registros,...

- a) **registros para recibir parámetros del procedimiento llamador**
- b) registros para almacenar matrices de enteros
- c) registros para enviar parámetros a otros procesos
- d) todas las respuestas son falsas

¿Cuál de las siguientes afirmaciones es cierta?

- a) **El modo de direccionamiento indirecto a través de registro es el modo más sencillo de direccionamiento a memoria.**
- b) Los modos de direccionamiento de una instrucción se especifican en ensamblador mediante las directivas.
- c) En el direccionamiento inmediato, la dirección efectiva se encuentra en un registro de uso general del procesador.
- d) En el direccionamiento a registro, el objeto direccionado es una constante contenida en la propia instrucción.

Una instrucción de lenguaje máquina debe:

- a) tener un número fijo de operandos, ya sean implícitos o explícitos.
- b) realizar una tarea sencilla y única, independiente de otras instrucciones.
- c) ser autocontenido e independiente de otras instrucciones o de su posición en el programa.
- d) **todas las respuestas anteriores son ciertas**

¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción mov bx, 5h?

- a) Direccionamiento inmediato
- b) Direccionamiento implícito
- c) **Direccionamiento directo a registro**
- d) Direccionamiento relativo a registro base

¿Cuál de las siguientes instrucciones no modifica necesariamente la secuencia de ejecución del programa?

- a) **JNE dir**
- b) JMP dir
- c) CALL dir
- d) RET

Cuando un operando se encuentra almacenado en un registro, se trata de un direccionamiento...

- a) **directo a registro**
- b) a registro inmediato

- c) a registro literal
- d) ninguno de los anteriores

¿Cuál de las siguientes parejas de microprocesadores representa mejor los conceptos RISC?

- a) Itanium, Alpha
- b) Pentium, Athlon
- c) PA-RISC, PowerPC
- d) **MIPS, SPARC**

Si AX = 0xFA50 y ejecutamos AND \$0xFF, %AX

- a) El registro AH se pone a FF
- b) El registro AL se pone a 0
- c) El registro AL se pone a FF
- d) **El registro AH se pone a 0**

Las arquitecturas de acumulador se caracterizan por:

- a) facilidad de generación de código eficiente para los compiladores.
- b) **diseño simple del procesador.**
- c) instrucciones muy largas.
- d) tráfico reducido entre procesador y memoria.

¿Cuál de las siguientes afirmaciones sobre compilación C->ASM es falsa?

- a) Puede que el compilador altere el orden del código C, apareciendo antes la traducción de sentencias C posteriores
- b) **Puede que el compilador altere por optimización el orden de los parámetros pasados en una llamada a función, si el marco de pila resultante es más eficiente**
- c) Puede suceder que varias sentencias C se traduzcan por una única instrucción ASM
- d) Puede que el compilador elimine por optimización construcciones C enteras (como un bucle for completo), si se conoce el resultado en tiempo de compilación

La instrucción setg %al:

- a) No cambia el contenido de AL
- b) Pone siempre AL a 1.
- c) Complementa AL si el resultado de la comparación anterior es A > B.
- d) **Pone AL a 1 en algunos casos.**

¿Cuál es la dirección física del operando fuente de la instrucción ADD AX, ETIQUETA siendo ETIQUETA = 7000h, CS = 1500h, DS = 4500h, e IP = 25h (la instrucción ocupa 3 bytes)?

- a) 4C000h

- b) 53000h
- c) 5C000h
- d) 15025h

Si queremos almacenar la palabra de 16 bits 8965h en memoria según "big-endian", quedará almacenada a partir de la posición 1000h como:

- a) en el byte 1000h se guarda 91h y en el 1001h A6h
- b) en el byte 1000h se guarda 65h y en el 1001h 89h
- c) **en el byte 1000h se guarda 89h y en el 1001h 65h**
- d) en el byte 1000h se guarda A6h y en el 1001h 91h

¿Qué tipo de direccionamiento se usa para el registro destino en la instrucción mov bx, 5h?

- a) Direccionamiento relativo a registro base
- b) Direccionamiento implícito
- c) Direccionamiento inmediato
- d) **Direccionamiento directo a registro**

Si %eax contiene x, ¿cuál de las siguientes instrucciones calcula x*5?

- a) leal (%eax,%eax,4),%edx
- b) leal 3(%eax,%eax,2),%edx
- c) leal 4(%eax,%eax),%edx
- d) leal 3(%eax,2),%edx

En el RISC-I, una ventana de registros contiene:

- a) registros para recibir parámetros del procedimiento llamador.
- b) registros para almacenar variables locales.
- c) registros para enviar parámetros a procedimientos.
- d) **todas las respuestas son ciertas.**

El microprocesador de la familia x86 usado en los primeros IBM PC originales fue el:

- a) 8088
- b) Pentium
- c) 8085
- d) 80486

Sobre el ensamblador:

- a) La calidad de un programa ensamblador afectará menos al tiempo de ejecución de los programas generados por él que la calidad de un compilador.
- b) Las etiquetas permiten que el programador especifique el destino de un salto de forma que éste no tenga que modificarse manualmente cuando el programa varíe de tamaño.
- c) El lenguaje ensamblador elimina la posibilidad de errores en la generación de la representación en lenguaje máquina de cada instrucción.

- d) Todas las respuestas son ciertas

¿Cuál de las siguientes listas está correctamente ordenada temporalmente?

- a) 486, 8086, Pentium MMX, Core 2, Pentium III, Pentium 4
- b) 8086, 486, Pentium MMX, Pentium III, Pentium 4, Core 2
- c) 8086, 486, Pentium III, Pentium MMX, Core 2, Pentium 4
- d) 486, 8086, Core 2, Pentium III, Pentium 4, Pentium MMX

En el direccionamiento indirecto a través de registro, la dirección efectiva...

- a) se encuentra en el registro de instrucción.
- b) se calcula como la suma del contenido de dos registros.
- c) se encuentra en una dirección de memoria.
- d) se encuentra en un registro general del procesador.

Una instrucción de salto si igual tiene que comprobar el valor de:

- a) el bit de signo
- b) el bit de acarreo
- c) el bit de cero
- d) los bits de signo y desbordamiento

En una máquina de 32 registros direccionables e instrucciones de 16 bits:

- a) No se pueden codificar a la vez 63 instrucciones de dos registros, una instrucción de 0 direcciones y 32 instrucciones de 1 registro.
- b) No se pueden codificar a la vez 62 instrucciones de dos registros, 32 instrucciones de 1 registro y 64 instrucciones de 0 direcciones.
- c) No se pueden codificar a la vez 64 instrucciones de 1 registro, 32 instrucciones de 2 registros y 32 instrucciones de 0 direcciones.
- d) No se pueden codificar a la vez 64 instrucciones de 0 direcciones, 63 instrucciones de 2 registros y 16 instrucciones de 1 registro.

Al compilar `popcount10`, se generan las instrucciones:

```
popcnt %rdx, %rdi  
popcnt %rsi, %rdx  
add %rdx, %rdi
```

El valor de la variable `result` al finalizar el bucle estará en el registro:

Usua Profeso

rio res

- a) %ea
x
- x** b) %rdi
- c) %edi
- d) %rdx

Puntuación: **-0,33**

Una placa madre de un 486 con un único SIMM de 30 contactos con 8 chips de 1M x 1...

Usua Profeso

rio res

- x**
- a) tiene 8 M palabras de 32 bits de memoria principal.
 - b) no funciona correctamente.
 - c) tiene 1 M palabras de 32 bits de memoria principal.
 - d) tiene 256 K palabras de 32 bits de memoria principal.

Puntuación: **-0,33**

¿En qué se diferencian las estrategias de mantenimiento de coherencia en memoria escritura directa y post-escritura?

Usua Profeso

rio res

- ✓**
- a) En cuándo tiene lugar la actualización
 - b) En cómo tiene lugar la actualización
 - c) Tanto en a) como en b)
 - d) Ni en a) ni en b)

Puntuación: **1,00**

¿Cuál es la característica tecnológica principal de la segunda generación de computadores?

Usuar Profesor

io es

- a) La gran integración de los circuitos (VLSI)
- b) Las válvulas
- c) Los transistores
- X d) Los circuitos integrados

Puntuación: **-0,33**

Al compilar `popcount10`, se generan las instrucciones:

```
popcnt %rdx, %rdi  
popcnt %rsi, %rdx  
add %rdx, %rdi
```

El valor de la variable `result` al finalizar el bucle estará en el registro:

Usuar Profesor

io es

- a) %ea
X
- b) %rdi
- c) %edi
- d) %rdx

Puntuación: **-0,33**

Un sistema no segmentado tarda 20 ns en procesar una tarea. La misma tarea puede ser procesada en un cauce (pipeline) de 4 segmentos con un ciclo de reloj de 5 ns. Cuando se procesan muchas tareas, la ganancia máxima de velocidad que se obtiene se aproxima a:

Usuar Profesor

io es

- a) 0,2
5
- b) 5
- c) 20
- ✓ • d) 4

Puntuación: **1,00**

15

Al compilar `popcount10`, se generan las instrucciones:

Elección
única

`popcnt %rdx, %rdi`
`popcnt %rsi, %rdx`
`add %rdx, %rdi`

El valor de la variable result al finalizar el bucle estará en el registro:

Usuar Profesor

io es

- a) %rdi
- b) %edi
- x** c) %rdx
- d) %ea

x

Puntuación: **-0,33**

2

Elección única

¿Cuál de las siguientes afirmaciones acerca de la memoria es cierta?

Usuario Profesores

- ✓ • a)

La memoria dinámica usa señales de control RAS# y CAS#.

b)

Las celdas de memoria estática tienen que ser constantemente refrescadas.

c)

La memoria dinámica se emplea en las caches L1 y L2.

d)

Las celdas de memoria estática están constituidas por un transistor y un condensador.

Puntuación: 1,00

1

Si val es una variable de tipo `unsigned int`, entonces la sentencia:

Elección
única

`val += (val >> 32);`

U Pro
su fes
ari ore
o s

- a Pone siempre val a 0.
)
- b Deja siempre val al mismo valor
) que tuviera antes de la sentencia.
- X** c Se traduce por una instrucción
) sal seguida de una suma.
d Se traduce por una instrucción
) sar seguida de una suma.

Puntuación: **-0,33**

5 Las instrucciones JB y JNAE provocan un salto si...

Elección única Usuar Profesor
 io es

- a) ZF != SF
- b) CF == 1
- c) ZF == 0
- X** d) SF == 1

Puntuación: **-0,33**

6 Con respecto a la memoria virtual, ¿cómo deben ser las páginas para evitar tener tablas de páginas de gran tamaño?

- a) Grandes
- b) Medianas
- c) Las tablas de páginas tiene un tamaño fijo e independiente del tamaño de las páginas
- X** d) Pequeñas

Puntuación: **-0,33**

La caché es gestionada por:

Usuar Profesor

o es



- - a) el programador
 - b) el sistema operativo
 - c) algoritmos hardware
 - d) ninguna es cierta

[T1.1]

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "little-endian", quedará almacenada a partir de la posición 0x8600 como:

Usuari Profesore

o s



- - a) $M[0x8600]=0x65$ y $M[0x8601]=0x89$
 - b) $M[0x8600]=0x69$ y $M[0x8601]=0x85$
 - c) $M[0x8600]=0x85$ y $M[0x8601]=0x69$
 - d) $M[0x8600]=0x89$ y $M[0x8601]=0x65$

Si queremos almacenar la palabra de 16 bits 0x8965 en una memoria de bytes según "big endian", quedará almacenada a partir de la posición 0x1000 como:

Usuari Profesores

o



- - a) $M[0x1000]=0x89$ y $M[0x1001]=0x65$

b) $M[0x1000]=0xA6$ y $M[0x1001]=0x91$

c) $M[0x1000]=0x65$ y $M[0x1001]=0x89$

d) $M[0x1000]=0x91$ y $M[0x1001]=0xA6$

En el 8086, la dirección física de la instrucción que se va a captar vendrá dada por:

Usuar Profesor

io es

a) DS +
desplazamiento

b) CS + IP

✓

• c) CS * 10h + IP

d) DS * 10h + IP

Puntuación: **1,00**

¿En qué se diferencian las estrategias de mantenimiento de coherencia en memoria escritura directa y post-escritura?

Usuar Profesor

io es

• a) En cuándo tiene lugar la
actualización

b) En cómo tiene lugar la actualización

c) Tanto en a) como en b)

x d) Ni en a) ni en b)

Puntuación: **-0,30**