

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 {
    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

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)

- 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 *microprogramación 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 *microprograma* 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
-

- 23.** ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?
- 2
 - 3
 - 4
 - 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?
- 4
 - 5
 - 6
 - 7
-
- 25.** ¿En qué tipo de transferencias es necesario establecer un periodo de tiempo máximo después del cual se considera que ha fallado?
- En las transferencias síncronas
 - En las transferencias asíncronas
 - En ambas
 - En ninguna
-
- 26.** ¿Qué técnica de E/S requiere menos atención por parte del procesador?
- E/S programada
 - E/S mediante acceso directo a memoria
 - E/S mediante interrupciones
 - 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"?
- Localidad espacial
 - Localidad secuencial
 - Localidad temporal
 - 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"?
- Con correspondencia directa
 - Totalmente asociativa
 - Asociativa por conjuntos
 - Ninguna de las anteriores
-
- 29.** ¿Cuál de las siguientes afirmaciones acerca de las memorias RAM dinámicas es cierta?
- Los datos permanecen en cada celda indefinidamente
 - Las celdas de almacenamiento son complejas
 - Las operaciones de lectura no son destructivas
 - 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?
- 31,25 microsegundos
 - 61 nanosegundos
 - 8192 milisegundos
 - 7,8125 microsegundos
-

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) ¿Cuál de las siguientes direcciones no está alineada a *double* (8-byte)?

- a) 1110110101110100)₂
- b) 1110110101101000)₂
- c) 1110110101110000)₂
- d) Todas están alineadas a *double*

2) ¿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

3) ¿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.

4) 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

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) 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) Hay registros que pueden ser modificados libremente por las subrutinas, y otros que deben ser tratados como lo anteriormente dicho: si se modifican se deben restaurar. Y también hay registros especiales.
- c) Hay registros modificables, otros que deben ser restaurados, y las subrutinas anidadas deben respetar los registros modificables que están en uso por otras subrutinas.
- d) Todos los registros pueden ser modificados libremente por todas las subrutinas.

7) 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).
 - 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.
-

8) 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 puntero es 4 bytes.
 - c) El tamaño de un `double` es 8 bytes.
 - d) El tamaño de un `short` es 2 bytes.
-

9) 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 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.
-

10) Se ha declarado en un programa C la variable `int val[5] = {1, 5, 2, 1, 3}`. ¿Cuál de las siguientes afirmaciones es FALSA?

- a) `val[1] == 1`
 - b) `&val[3] == val+3`
 - c) `sizeof(val) == 20`.
 - d) Todas son ciertas.
-

11) ¿Qué tipo de operaciones de E/S consume menos tiempo del procesador?

- a) E/S programada
 - b) E/S mediante interrupciones
 - c) E/S mediante DMA
 - d) Todos consumen el mismo tiempo del procesador
-

12) ¿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
-

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

- a) MAR y ACUMULADOR
- b) IR y ACUMULADOR

- c) MBR y MAR
 - d) MBR y PC
-

14) ¿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) Entrada desde un escáner
-

15) ¿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) No es necesario sincronizar el procesador con los dispositivos de E/S
-

16) En la ejecución de una instrucción...

- a) el registro de instrucción se va incrementando para apuntar a la siguiente instrucción
 - b) la ALU realiza las operaciones aritméticas y lógicas
 - c) la UC activa las señales de control que envía por el bus de direcciones
 - d) siempre se altera el registro de estado
-

17) 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) 1
 - b) 9
 - c) 8
 - d) 4
-

18) 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
-

19) ¿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.
-

20) 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 en general 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.
-

21) Una posible codificación en microinstrucciones de la instrucción CALL X es:

- a) $SP=SP-1$; $m[SP]=PC$; $PC=PC+1$
 - b) $SP=PC-1$; $m[SP]=PC$; $PC=X$
 - c) $PC=X$; $SP=SP-1$; $m[SP]=PC$
 - d) $SP=SP-1$; $m[SP]=PC$; $PC=X$
-

22) Respecto al sistema de Entrada / Salida, ¿cuál de las siguientes afirmaciones es errónea?

- a) Un módulo de E/S se encarga de la comunicación con el procesador.
 - b) Un protocolo sirve para “ponerse de acuerdo” en cosas como velocidad, paridad, nº de bits, etc.
 - c) La mayoría de los periféricos trabajan a velocidad muy superior al procesador; por eso es necesario sincronizar.
 - d) El procesador se comunica con el periférico por medio del controlador y de software de E/S.
-

23) ¿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
-

24) ¿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
-

25) ¿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
-

26) Cada celda de un chip de memoria DRAM de $1M \times 1$, 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
-

27) 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
-

28) 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) 25 ns
 - b) 34 ns
 - c) 88 ns
 - d) 112 ns
-

29) En una jerarquía de memoria, a medida que nos alejamos de la CPU:

- 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
 - d) el coste por byte aumenta
-

30) 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?

- a) Más de 8 veces.
 - b) La caché de postescritura no puede ser más eficiente que la de escritura inmediata.
 - c) La caché de postescritura siempre será más eficiente que la de escritura inmediata.
 - d) Depende de la tasa de aciertos.
-

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

- En una CPU de 32bits con memoria de bytes, el problema es que...
 - No tiene sentido, un registro no cabría en memoria
 - No hay problema, cuando se salva un registro a memoria se escribe en la posición deseada
 - Hay que usar 4 instrucciones de lectura (o escritura) para leer (o escribir) un registro completo
 - Hay que respetar el ordenamiento de bytes y reglas de alineamiento con que se diseñó la CPU
- 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...
 - Hay una palabra de 16bit big-endian con valor 1 en la posición 0
 - Hay una palabra de 16bit little-endian con valor 254 en la posición 3
 - Hay una palabra de 32bit little-endian con valor -1 en la posición 4
 - Todas las respuestas anteriores son incorrectas
- ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):
 - movw %dx, (%eax)
 - pushb \$0xFF
 - movswl (%eax), %edx
 - movzbl %dl, %eax
- La instrucción xor \$3, %eax tiene como resultado:
 - Poner a 0 los últimos 3 bits del registro EAX
 - Cambiar 0<->1 (complemento a 1 de) los últimos 2 bits del registro EAX
 - Poner a 1 el último bit del registro EAX
 - Ninguno de los anteriores resultados
- De entre las siguientes construcciones de flujo de control en lenguaje C, la que se traduce más directamente a lenguaje ensamblador es...
 - El bucle for
 - El bucle while
 - El bucle do-while
 - La selección switch-case
- GCC/Linux IA32 resuelve el ajuste de marco de pila mediante las instrucciones:
 - 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. $((*(x + 8)) + 28)$
 - b. $((*(x + 2)) + 7)$
 - c. $((*(x + 2) + 7)$
 - d. $(((*x) + 2) + 7)$
-

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

d. Se pone a 1 cuando el resultado es positivo

15. ¿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 se emplean tanto para memoria como para E/S
 - c. La E/S independiente facilita la protección
 - d. La E/S en memoria es mucho más rápida que la E/S independiente
-

16. La técnica de sondeo, escrutinio o "polling"...

- a. Se utiliza para identificar la fuente de una interrupción
 - 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. 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:

- a. E/S por nivel
 - b. E/S por flanco
 - c. Acceso directo a memoria (DMA)
 - d. Acceso indirecto a memoria (IMA)
-

18. ¿Cuál de las siguientes afirmaciones acerca del concepto de interrupción es cierta?

- a. Es una bifurcación normalmente externa al programa en ejecución
 - b. Su objetivo es incrementar el ancho de banda con el dispositivo
 - c. Solicita que el procesador se aisle de los buses
 - d. 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:

- a. `mov %esp, %ebp`
 - b. `push %ebp`
 - c. `push %ebx`
 - d. `pop %ebx`
-

20. En cdecl/x86, ¿cuál de los siguientes registros tiene que ser guardado por la función llamada si es alterado por ésta?

- a. `eax`
 - b. `ebx`
 - c. `ecx`
 - d. `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:

- 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
-

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.

- a. 2 ns
 - b. 3 ns
 - c. 4.5 ns
 - d. 9 ns
-

23. ¿Cuántas patillas de dirección tiene una memoria DRAM de 1G palabra, siendo la longitud de palabra de 16 bits?

- a. 20
 - b. 16
 - c. 30
 - d. 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
-

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 0x0000001f fffffe0 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.

- 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

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. ECX
- d. EDX

8. 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

9. ¿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. RBX, RSI, RDI
- c. RSP, RBP
- d. CL, DX, R8d, R9

10. 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 sobrescribiría los valores escritos en pila)
- b. 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:

- 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

12. 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

13. 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 EPROM (Electrically Progr. ROM) se puede grabar eléctricamente, sin fundir contactos, pero no se puede borrar
- d. Una EEPROM (Erasable EPROM) se puede grabar (eléctricamente), y borrar (usando rayos ultravioleta)

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. $N \cdot C / F$
 - 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`, rellenando 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 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 0x0 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?

- a. 2 bytes
- b. 4 bytes
- c. 6 bytes
- d. 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.

- a. leal 12(%ecx),%eax
- b. leal (%ecx,%ebx,4),%eax
- c. movl (%ecx,%ebx,4),%eax
- d. movl 8(%ecx,%ebx,2),%eax

8. alguna de las siguientes afirmaciones sobre sistemas Linux x86-64 **no** es cierta

- a. %rax se usa para devolver los valores de retorno de funciones
- b. %eax y %ebx pueden usarse como en un sistema IA-32
- c. Todos los argumentos de función se pasan a través de la pila
- d. %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?

- a. 0x7ffffeffff8
- b. 0x7fffffff0000
- c. 0x7fffffff0004
- d. 0x7fffffff0008

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
```

- a. 08048391 + 12 = 08048403
- b. 08048391 - 12 = 08048379
- c. 0804837d
- d. 0804837f

11. En la técnica de salto retardado:

- a. El compilador puede reorganizar el código para rellenar 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.

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?

- a. La ganancia de velocidad (aceleración) con la segmentación es de 5.
- b. La preparación completa de cada tarta sigue requiriendo 50 s (igual que con una sola persona empaquetando las tartas).
- c. Cada 50 s saldrá una nueva tarta empaquetada, el mismo tiempo que cuando no había cadena de empaquetado.
- d. 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?

- a. Decodificación de las instrucciones.
- b. Secuenciamiento de las instrucciones.
- c. Generación de las señales de control que provocan la ejecución de cada instrucción.
- d. Cálculo de operaciones de coma flotante.

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:

- a. daisy-chain
 - b. interrupción
 - c. polling
 - d. 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?

- a. 1%
 - b. 10%
 - c. 50%
 - d. 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?

- a. 0,75
 - b. 0,9
 - c. 0,95
 - d. 0,8
-

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

- a. 0xFFFFFFFF0C
 - b. 0xFF0C
 - c. 0xFFFFFFFFF4
 - d. 0xFFF4
-

18. Cuando se produce una interrupción hardware...

- a. Se salta al principio del programa actual.
- 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 guarda el estado y se ejecuta la rutina de interrupción asociada.
-

19. Para realizar la microoperación $MAR \leftarrow PC$, habrá que activar:

- a. LdPC y EnMAR
 - b. EnPC y LdMAR
 - c. LdPC y LdMAR
 - d. 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?

- a. 64
 - b. 1024
 - c. 8192
 - d. 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`?

- a. 2^{12}
 - b. 2^{14}
 - c. 2^{16}
 - d. 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?

- a. 6 bits
 - b. 13 bits
 - c. 29 bits
 - d. 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?

- a. 256 K x 1
 - b. 64 K x 4
 - c. 32 K x 8
 - d. 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)

- a. [5...0]
 - b. [11...6]
 - c. [5...3]
 - d. [8...6]
-

25. ¿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.
-

26. ¿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.
-

27. ¿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.
-

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.

- a. 2 ns.
 - b. 3 ns.
 - c. 4.5 ns.
 - d. 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?

- a. 20
 - b. 16
 - c. 30
 - d. 15
-

30. ¿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.
-

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 acoplo, 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:

- 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. 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
```

- a. CF
 - b. OF
 - c. SF
 - d. ZF
-

20. 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. 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 **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
-

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, 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
 - 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 x 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 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 específica 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 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

- 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?
 - 0
 - 1
 - 1
 - 2
- Según la clasificación m/n, las máquinas de acumulador son de tipo
 - 0/0
 - 1/1
 - 1/2
 - 2/2 ó 2/3
- Según la clasificación m/n, las máquinas con arquitectura R/R son de tipo
 - 0/0
 - 0/x con x=2,3
 - x/0 con x=2,3
 - x/x con x=2,3
- El bus del sistema es
 - el que conecta las distintas partes del sistema: UC, ALU, E/S, M
 - en un sistema con bus único, todo el bus salvo la parte relacionada con E/S (SATA, GPU, USB, Ethernet, etc)
 - en un sistema con buses separados, el que conecta el sistema E/S con el resto
 - el que conecta CPU-M, ya sea un sistema con bus único o con múltiples buses
- 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
 - 0x00 a 0x0f y 0x80 a 0x8f
 - 0x00 a 0x1f
 - 0x70 a 0x7f y 0xf0 a 0xff
 - 0xe0 a 0xff
- 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
 - cómo de mejor es un procesador frente a otro, conociendo las prestaciones de las respectivas UC, ALU, E/S y M
 - 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

- 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 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
-

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 n° 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 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

- Si almacenamos según el criterio little-endian la palabra de 64 bits 0xFACEB00C a partir de la dirección 0xCAFEBAFE, el byte 0xCE quedará almacenado en la dirección:
 - 0xCAFEBAF1
 - 0xCAFEBAF0
 - 0xCAFEBAF5
 - 0xCAFEBAFE
- ¿Qué novedad se desarrolló en la tercera generación de computadores?
 - Los circuitos integrados
 - Los primeros lenguajes de programación de alto nivel
 - Los microprocesadores CISC
 - Los microprocesadores RISC
- En X86-64, el registro contador de programa se denomina:
 - RIP
 - EIP
 - IP
 - R15
- La primera letra (l) de la instrucción lea:
 - indica que la instrucción usa ordenación de bytes little-endian
 - indica que la instrucción trabaja con un operando destino de 32 bits (long word)
 - indica que la instrucción afecta a los 16 bits menos significativos del operando destino (low word)
 - forma parte del nemotécnico de la instrucción
- ¿Cuál es la diferencia entre las instrucciones mov y lea?
 - lea accede a la posición indicada, mientras que mov no lo hace
 - mov accede a la posición indicada, mientras que lea no lo hace
 - lea puede usarse para copiar un registro a otro, mientras que mov no
 - mov puede usarse para copiar un registro a otro, mientras que lea no
- La instrucción not:
 - realiza el complemento a dos
 - realiza el complemento a uno (cambiar unos por ceros y ceros por unos)
 - realiza la operación no-or (or negada)
 - realiza un salto condicional si negativo
- La instrucción JGE / JNL provoca un salto si...
 - SF = 1
 - CF = 1
 - SF = 0
 - OF = SF
- Un overflow nunca puede ocurrir cuando:
 - se suman dos números positivos
 - se suman dos números negativos
 - se suma un número positivo a un número negativo
 - se resta un número positivo de un número negativo
- 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”?

- 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

- 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 ahorramos 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 rellenar 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 los 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 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

- 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:
 - arquitectura del computador
 - conjunto de componentes físicos del computador
 - organización del computador
 - repertorio de instrucciones máquina
- ¿Cuál de las siguientes afirmaciones es cierta?
 - la arquitectura Von Neumann de los computadores tradicionales consiste en tener almacenados los datos separados de las instrucciones en memorias distintas
 - 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
 - la unidad de control necesita como entrada el registro contador de programa para saber cuál es la instrucción que debe ejecutar a continuación
 - el registro de direcciones de memoria es un registro de propósito general que puede contener tanto direcciones como datos
- 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
 - 4
 - no se puede saber, faltan datos
 - ninguna de las anteriores
- 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?
 - 0xFACEB00C
 - 0xDEADBEEF
 - 0xCAFEBABE
 - 0xABADF00D
- En una arquitectura de acumulador, la instrucción LOAD X:
 - transfiere el contenido del registro X a la memoria
 - suma M(X) al acumulador
 - transfiere el contenido del acumulador a la posición de memoria X
 - transfiere el contenido de la posición de memoria X al acumulador
- Una instrucción máquina del tipo "Add M,R" podría formar parte del repertorio de
 - una máquina pila
 - una máquina de acumulador
 - una máquina con arquitectura R/R
 - una máquina con arquitectura M/M

7. ¿Cuál de los siguientes no es un modo de direccionamiento IA-32?

- a. Registro
 - b. Memoria
 - c. Cache
 - d. Inmediato
-

8. Un bus se compone de:

- a. líneas de datos y líneas de dirección
 - b. líneas de alimentación
 - c. líneas de estado y líneas de control
 - d. líneas de control/estado, líneas de dirección y líneas de datos
-

9. ¿Cuál de los siguientes no es un tipo de bus?

- a. Secuencial
 - b. Paralelo
 - c. E/S
 - d. 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?

- a. 10
 - b. 16
 - c. 32
 - d. 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...

- a. aritmética
 - b. geométrica
 - c. armónica
 - d. ponderada
-

12. El primer computador electrónico basaba su funcionamiento en:

- a. tubos de vacío
 - b. circuitos integrados LSI
 - c. amplificadores operacionales
 - d. núcleos de ferrita
-

13. En Linux IA-32, si gcc usa la instrucción leave se puede asegurar que en ese punto del programa

- a. correspondería emitir la secuencia de salida pop/ret, pero leave hace lo mismo y ocupa menos espacio
 - b. ya no hay registros salva-invocado que recuperar
 - c. ya no hay variables locales que destruir
 - d. 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...

- a. dos instrucciones mov
 - b. una instrucción mov y una instrucción lea
 - c. 3 mov, no menos (se le llama "intercambio circular")
 - d. 4 mov, no menos (debido a la arquitectura R/M)
-

15. Respecto a registros base e índice en IA-32, la excepción es que

- a. EBP no puede ser registro base
 - b. EBP no puede ser registro índice
 - c. ESP no puede ser registro base
 - d. ESP no puede ser registro índice
-

16. El registro SP / ESP / RSP...

- a. es un registro transparente al usuario y contiene la instrucción que se está ejecutando
 - b. es un registro de propósito específico y contiene la dirección de la cima de la pila
 - c. es un registro transparente al usuario y contiene la dirección de memoria a la que se está accediendo
 - d. 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

- a. los enteros largos (long) pasan de 32 a 64 bits
 - b. los punteros (void*) pasan de 32 a 64 bits
 - c. el tipo double pasa de 4 B a 8 B
 - d. 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?

- 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 acoplo (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 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

- ¿Cuál de las siguientes afirmaciones sobre el benchmark SPEC CPU es falsa?
 - La última versión es SPEC CPU2006 V1.2 de 2011
 - Se cronometran unos 12 tests de enteros (CINT2006) y unos 17 tests de punto flotante (CFP2006)
 - 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
 - El resultado final es la media aritmética de las (12 ó 17) velocidades, bien sea de enteros ó de punto flotante (SPECint2006 ó SPECfp2006)
- ¿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?
 - 2ª generación (1955-65)
 - 3ª generación (1965-75)
 - 4ª generación (1975-85)
 - esas innovaciones se repartieron a lo largo de varias generaciones, no sólo una
- Respecto a tamaños de tipos integrales en x86 y x86-64, la excepción es que
 - int pasa de 4 B (x86) a 8 B (x86-64)
 - long int pasa de 4 B a 8 B
 - long long int pasa de 4 B a 8 B
 - ninguna de las anteriores
- Con el repertorio IA32, para sumar %eax y %ebx dejando el resultado en %ecx se podría hacer lo siguiente:
 - lea %eax, %ebx, %ecx
 - lea (%eax, %ebx, 1), %ecx
 - lea %ecx, [%eax, %ebx]
 - lea %ecx, %ebx, %eax
- Cuál de las instrucciones máquina siguientes es incorrecta en x86-64:
 - testl %edx, %edx
 - movl %r8, %eax
 - movl (%rdi,%rcx,4), %edx
 - addq \$1, %rcx
- 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:
 - xorl %ebx,%eax
 - xorl %eax,%ebx
 - andl %ebx,%eax
 - testl %eax,%ebx
- Para poner a 1 el bit 5 del registro %edx sin cambiar el resto de bits podemos usar la instrucción máquina:
 - and \$32, %edx
 - and \$0x5, %edx
 - or \$0b101, %edx
 - or \$0x20, %edx
- 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

-
11. Respecto a requisitos de alineamiento de structs en gcc/IA32 x86 y x86-64, alguna de las siguientes afirmaciones es falsa

- 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 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.

- c. Las únicas instrucciones en las que algunas de sus fases de ejecución conllevan 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?

- a. 20000 bits
 - b. 21600 bits
 - c. 22400 bits
 - d. 30000 bits
-

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

- a. Se utiliza para realizar un microsalto condicional en función del registro de estado.
 - b. Realiza una llamada a una microsubrutina.
 - c. Salta a una dirección de memoria de control que depende de la instrucción máquina actual.
 - d. 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

- 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
 - d. 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

- a. idealmente, con el segmentado se intenta ejecutar una instrucción por ciclo, y con el superescalar más de una por ciclo (al combinarlo con segmentado)
 - b. en cualquier procesador resulta ventajoso usar una cola de instrucciones, pero es más importante para uno segmentado (fundamental) que para uno superescalar (conveniente)
 - c. por definición, un procesador superescalar debe tener varias unidades funcionales (más de una)
 - d. 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:

- a. La controladora o interfaz contiene los puertos necesarios para utilizar el dispositivo
 - 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
-

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. 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.
- c. 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. 0xFBBF
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 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

- 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?
 - El desplazamiento D puede ser una constante literal (1, 2 ó 4 bytes)
 - EBP no se puede usar como registro base
 - ESP no se puede usar como registro índice
 - El factor de escala S puede ser 1, 2, 4, 8
- La extensión de signo a m bits de un número original N de n bits, con $m > n$, consiste en:
 - Realizar la operación $2^m - N$
 - Realizar la operación $2^m - N - 1$
 - Incrementar la cantidad de bits a m preservando el signo y el valor del número.
 - Incrementar la cantidad de bits a m rellenando con unos por la izquierda.
- En IA32, ¿cuál de los siguientes fragmentos de programa tiene un efecto sobre los flags distinto al resto?
 - `sub %edi, %edi`
`adc $0xFFFFFFFF, %edi`
 - `mov $-1, %edi`
 - `mov $-1, %edi`
`add $0, %edi`
 - `mov $0, %edi`
`sub $1, %edi`
- Si `%rsp` vale `0xdeadbeefdeadd0d0`, ¿cuál será su nuevo valor después de que se ejecute `pushq %rbx`?
 - `0xdeadbeefdeadd0d4`
 - `0xdeadbeefdeadd0d8`
 - `0xdeadbeefdeadd0cc`
 - `0xdeadbeefdeadd0c8`
- ¿Cómo se devuelve en ensamblador x86-64 Linux gcc el valor de retorno de una función `long int` al terminar ésta?
 - La instrucción `RET` lo almacena en un registro especial de retorno.
 - Por convención se guarda en `%eax`.
 - Se almacena en pila justo encima de los argumentos de la función.
 - Ninguna de esas formas es la correcta.
- Comparando las convenciones de llamada de gcc Linux IA32 con x86-64 respecto a registros
 - En IA32 `%ebx` es salva-invocante, pero en x86-64 `%rbx` es salva-invocado
 - En IA32 `%ecx` es salva-invocante, y en x86-64 `%rcx` es salva-invocante también
 - En IA32 `%esi` es salva-invocado, y en x86-64 `%rsi` es salva-invocado también
 - En IA32 `%ebp` es especial (marco de pila), y en x86-64 `%rbp` también
- 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

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. 4 bits
 - b. 5 bits
 - c. 16 bits
 - d. 17 bits
-

11. 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
-

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
-

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
ldx 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:

- a. 8 niveles de prioridad
 - b. 16 niveles de prioridad
 - c. 24 niveles de prioridad
 - d. Ninguna de las anteriores es cierta
-

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. Respecto al refresco de memorias DRAM, ¿cuál de las siguientes afirmaciones es falsa?

- a. Una operación de refresco consiste en dar un impulso /CAS junto con una dirección de columna.
 - b. Los chips DRAM refrescan automáticamente la fila accedida en cualquier ciclo de lectura o escritura.
 - c. Se precisa una circuitería auxiliar, externa al chip DRAM o integrada en él, que produzca ciclos de refresco.
 - d. 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:

- a. La capacidad (tamaño) s_i del nivel i .
 - b. La estrategia de administración de memoria.
 - c. La unidad de la transferencia de información x_i entre el nivel i y el $i+1$.
 - d. El ancho de banda b_i del nivel i .
-

27. La política de correspondencia de una memoria cache con 1 único conjunto es:

- a. Directa
 - b. Totalmente asociativa
 - c. Asociativa por conjuntos con una única línea
 - d. Asociativa por conjuntos de una única vía
-

28. La política de correspondencia de una memoria cache con la mitad de conjuntos que líneas es:

- a. Asociativa por conjuntos de 2 vías
 - b. Totalmente asociativa de media vía
 - c. Asociativa por conjuntos con 2 líneas
 - d. Directa con 2 líneas
-

29. Para construir una DRAM de 4GB con pastillas de 512Mx4bit hacen falta

- a. 8 pastillas
 - b. 16 pastillas
 - c. 32 pastillas
 - d. 64 pastillas
-

30. Para diseñar una memoria con ancho de palabra $k \times m$ (y mismo n° 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 se conecta a $A_0 \dots A_{m-1}$, el 2º a $A_m \dots A_{2m-1}$, etc
-

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
cmpl    $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
cmovle $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

18. ¿Cuál de las siguientes afirmaciones es ***FALSA***?

- a. La operación de lectura de una celda DRAM es destructiva
 - b. Las memorias DRAM son en general más lentas que las SRAM
 - c. Una celda DRAM no pierde la información al desconectar la alimentación
 - d. Las memorias DRAM presentan generalmente una capacidad de almacenamiento mayor que las SRAM
-

19. La memoria DRAM:

- a. Se inventó en la década de los 2000
 - b. Necesita 6 transistores por cada celda
 - c. Se denomina dinámica porque su contenido puede alterarse, al contrario que la SRAM
 - d. 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
```

- a. $08048391 + 12 = 08048403$
 - b. $08048391 - 12 = 08048379$
 - c. 0804837d
 - d. 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; };
```

- a. 18 B
 - b. 20 B
 - c. 22 B
 - d. 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***)

- a. codificación
- b. solapamiento
- c. 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

- a. 4 bits
 - b. 5 bits
 - c. 16 bits
 - d. 17 bits
-

24. Motivos que impiden que la ganancia (aceleración) de un cauce segmentado sea ideal (señale la respuesta ***FALSA***)

- a. registros de acoplo (coste de la segmentación)
 - b. fragmentación desigual (duración desigual de etapas)
 - c. riesgos (*hazards*)
 - d. 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?

- a. 2 ns
 - b. 3 ns
 - c. 4,5 ns
 - d. 9 ns
-

26. 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
-

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
-

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

- ¿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)
 - 2.147.483.647
 - 2.147.483.648
 - 4.294.967.295
 - 4.294.967.296
- ¿Cómo se representa el valor -1 como entero con signo en 14 bits?
 - 0xFFFF
 - 0x3FFF
 - las respuestas anteriores no son válidas porque usan hexadecimal; habría que usar binario
 - no se puede porque 14 no es múltiplo de 4
- ¿Cuál de las siguientes no es una unidad de la arquitectura Von Neumann?
 - Unidad central de proceso
 - Memoria principal
 - Sistema de entrada/salida
 - Núcleo del sistema operativo
- ¿Cuál de las siguientes afirmaciones es verdadera?
 - La arquitectura Von Neumann en la que se basan los computadores tradicionales consiste en tener los datos separados de las instrucciones en memorias distintas.
 - El registro de estado es un registro transparente al usuario, ya que éste no puede utilizarlo en las instrucciones máquina.
 - El registro de instrucción es un registro transparente al usuario, ya que éste no puede utilizarlo en las instrucciones máquina.
 - La unidad de control necesita como entrada el registro contador de programa, para saber cuál es la instrucción que debe ejecutar a continuación.
- ¿Qué es el lenguaje máquina?
 - Conjunto de datos binarios que representan señales eléctricas internas de la unidad de control de un microprocesador.
 - Conjunto de sentencias en un lenguaje escrito que se utilizan para generar programas codificados en lenguaje ensamblador.
 - 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.
 - Conjunto de instrucciones en formato binario que entiende un determinado procesador.
- ¿Cuál de los siguientes elementos no forma parte de la Arquitectura del Repertorio de Instrucciones (ISA)?
 - 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;
```

```
float bit2float(unsigned u) {
    bit_float_t arg;
    arg.u = u;
    return arg.f;
}
```

¿Cuál afirmación es verdadera?

- Si asignamos `f1=bit2float(u1)`; entonces `f1== 2147483648.00`
- Si asignamos `f1=bit2float(u1)`; entonces `f1== -0.0`
- Si asignamos `f2= (float)u1` ; entonces `f2== 4294967296.00`
- 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:

- Intercambiar el elemento 0 del vector con el 99, el 1 con el 98, el 2 con el 97 y así sucesivamente.
- Intercambiar el elemento 0 del vector con el 1, el 1 con el 2, el 3 con el 4 y así sucesivamente.
- 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.
- 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***)

- registros de acoplo (coste de segmentación)
- fragmentación desigual (duración de etapas)
- riesgos (hazards)
- 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?

- 2 ns
- 3 ns
- 4.5 ns
- 9 ns

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 (opcionalmente, 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 $D_{n-1}-D_0$.
 - c. Direcciones $A_{n-1}-A_0$.
 - 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.
-

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

2. Escoger de entre las 4 operaciones la de mayor valor que pueda calcularse con enteros de 4B con signo sin problemas

- a. 100.000.000 + 100.000.000
- b. 300.000.000 + 300.000.000
- c. 1.000.000.000 + 1.000.000.000
- d. 3.000.000.000 + 3.000.000.000

3. 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. $RCX \neq -1$, **CF** mantiene su valor
- c. $RCX = -1$, **CF** = 1
- d. no se puede marcar ninguna de las opciones anteriores

4. El registro **RAX** contiene el número binario 111111110111000010. ¿Cuál será su contenido tras ejecutar la instrucción **sar \$1,%ax**?

- a. 0x3fde1
- b. 0x7fde1
- c. 0xffde1

d. 0xffffffffffffde1

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 xxxxxx
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**

- 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:

- 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 segundos 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 ($0 \dots w-1$) (los menos significativos)
 - b. bits $w \dots w+c-1$ (con $c=b-v$)
 - c. bits $w \dots w+c-1$ (siendo $2^c = n^\circ$ conjuntos)
 - d. bits $b \dots b+c-1$ (siendo $2^c = n^\circ$ 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
-