

Nombre:		
DNI:		Grupo:

Test de Teoría (puntuaría sobre 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
a	c	c	a	d	d	a	c	b	b	c	a	c	c	a	a	d	d	c	a	b	d	a	a	b	b	b	b	b	b

1. Cuando en C se compara un valor sin signo con otro con signo, la operación se realiza pasando previamente todos los valores a sin signo. Si hay que extender previamente el número de bits, la extensión se hace con o sin signo dependiendo del tipo fuente. ¿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");
```

- No imprime nada
- Imprime "Hello"
- Imprime "World"
- Imprime "HelloWorld"

Re: En el primer if, (short) 0xFFFF es -1, se extiende a 4B (-1=0xFFFFFFFF) pero como x no tiene signo se interpreta como 4G-1, x es mucho menor y no se imprime Hello. En el segundo if, z=-1 se interpreta como 4G-1 (no hay nada que extender) y tampoco se imprime World

2. ¿Cuál de las siguientes instrucciones convierte %eax = 5 * %eax?

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

- Ninguna
- Sólo 1
- Sólo 2
- Ambas 1 y 2

Re: la primera mueve adonde apunta 5*eax (el contenido de 5*eax) a eax

3. 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];
```

- i externo, j central, k interno (el orden en que están ahora)
- j externo, k central, i interno
- k externo, i central, j interno
- El orden de los bucles no afecta a la localidad

Re: row-major order hace que corra más rápido (estén contiguos los datos en) la dimensión Z, luego Y y luego X, controladas por las

variables j,i,k, luego queremos interno j, medio i, externo k

4. ¿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. $(x | 0x3f)$
-

Re: $31=0x1f$ así que un múltiplo de $32=0x20$ no tiene activado ningún bit de los de $0x1f$, es decir $(x \& 0x1f) == 0$, o sea (a)

5. 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. 0x7ffffffeffff8
 - b. 0x7fffffff0000
 - c. 0x7fffffff0004
 - d. 0x7fffffff0008
-

Re: retq incrementa rsp++, en 8B=8 posiciones

6. 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
-

Re: la posición (x) es justamente el valor de eip que se usa para direccionamiento relativo a contador de programa en el je previo, así que $x+0x12=0x08048391$, pero la resta (b) está hecha “en decimal”, (d) está bien

7. 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.
-

Re: ver Tema3 transparencia tr.36 (y 37)

8. 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
-

Re: ver T5 tr.36-38,40,58,61,63-64,97

9. 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%
-

Re: ver T5 tr.102: la ISR gasta $500 \times 2 = 1000$ ciclos, cada segundo hay $1E5$ IRQs = $1E5 \times 1E3 = 1E8$ ciclos de ISR por cada $1E9$ (1GHz) ciclos CPU -> $1E8/1E9 = 10\%$

10. 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
-

Re:ver T6 tr.24-27,119-120: $T=A(2)+(1-A)(80+2)=10=2A-82A+82=82-80A \rightarrow A=(82-10)/80=0.9=90\%$

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

- a. 0xFFFFFFFF0C
- b. 0xFF0C
- c. 0xFFFFFFFFF4
- d. 0xFFFF4

Re:12=0xc compl.1=0xffffffff3 compl.2 0x...f4

12. 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}

Re:ver T1 tr.30-31: 48->6b/reg ($2^5=32$), 200->8b/codop ($2^7=128$), quedan 32-8-6-6=12b para la dirección

13. ¿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

Re:ver T6 tr.135-138, sobre todo tr.137: asumiendo memoria de bytes, w=6, N=42, 8MB= 2^{23} B= 2^{17} lineas= 2^{13} conj: c=13, y entonces N-c=29

14. 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

Re:ver T6 tr.58: 256Kb= 2^{18} b, 2^{15} posiciones=32Kpos, $2^{18}/2^{15}=2^3$ b/pos -> 32Kx8, 32KB

15. 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]

Re:ver T6 tr.130,133,137,141, pero $64=2^6$

16. ¿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.

Re:ver T6 tr.49-50,59

17. ¿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

Re:ver T6 tr.47-48,61-62, 16b para despistar, $1G=2^{30}=2^{15} \times 2^{15}$ filasxcolumnas

18. ¿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
-

Re:ver T2.1 tr.17-18,42-43, T2.4 tr.4-5, T2.5 tr.7.

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

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

Se declara una variable global my_node como:

```
struct NODE my_node;
```

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

- a. 0x601050
- b. 0x601054
- c. 0x601055
- d. Ninguna de las anteriores

Re:desplazamientos en RECORD: 0, 8, 12, tam16. NODE: 0, 8, 24, tam32. Así r.tag[1] es desplazamiento $8+12+1=21=0x15$

20. En la pregunta anterior, ¿cuál es el tamaño de my_node en bytes?

- a. 32
- b. 40
- c. 28
- d. Ninguno de los anteriores

Re:cálculo hecho arriba, 0, 8, 24, tam32

21. ¿Cuántas señales de control se necesitan como mínimo para implementar un sistema de gestión de interrupciones?

- a. 1
- b. 2
- c. 3
- d. 4

Re:ver T5 tr.55,60-67,72-74

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

- a. 4
- b. 5
- c. 6
- d. 7

Re:ver T5 tr.73: 4bits msk+2bits lvl+1bit BGII

23. Cada celda de un chip de memoria DRAM de 1M x 1, organizada en una matriz de 512 filas x 2048 columnas, necesita ser refrescada cada 16 ms. ¿Cada cuánto tiempo ha de realizarse una operación de refresco en el chip?

- a. 31,25 microsegundos
- b. 61 nanosegundos
- c. 8192 milisegundos
- d. 7,8125 microsegundos

Re:ver T6 tr.46,54-57. $16\text{ms}/512\text{filas} = 2^4/2^9 = 1/2^5 \sim 1/30 = 0.033\text{ms/fila} (=31.25\mu\text{s/fila})$

24. 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

Re:ver T3 tr.50-52. $16000 \times 250 = 16\text{E}3 \times 1\text{E}3/4 = 4\text{E}6 = 4000000$ bits microprograma. 447 uinstr. necesita 9bits ($2^8=256$). $16\text{E}3 \times 9 = 144\text{E}3$ bits μprog , $447 \times 250 = 447 \times 1\text{E}3/4 = 111750$ bits nprog. Suma 255750. Ahorro $4000000 - 255750 = 3744250$.

25. 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

Re:ver T4 tr.15-19. $T_c = T/n$

26. 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. 37 ns
- c. 88 ns
- d. 112 ns

Re:ver T6 tr.24-27,119-120: $T = 0.75 \times 12 + 0.25 \times (100 + 12) = 9 + 28 = 37$

27. ¿Cuál de las siguientes instrucciones es errónea? (sale mensaje de error al intentar ensamblar):

- a. `movw %dx, (%eax)`
- b. `pushb $0xFF`
- c. `movswl (%eax), %edx`
- d. `movzbl %dl, %eax`

Re:sale suffix or operands invalid for push, no se puede meter 1B suelto en pila (desalinea)

28. ¿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

Re:ver T6 tr.129-131, sobre todo 130: $w=4$, $N=12$, $256_{\text{pal}}/16_{\text{pal}}/\text{bl}=16_{\text{blq}} \rightarrow n=4$, $N-n=8$ bits etiqueta. No puede ser 2bl en mismo marco, es decir, mismo campo "marco" (3º dígito hex) y distinta etiq (1º-2º dígito hex): a,c no puede ser, b sí puede ser (marco A etiq AB, distintas palabras B,C del mismo bloque)

29. 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

Re:ver T6 tr.51-53. $256=2^8$, últimos 2 dígs hex indican palabra dentro de la página, 0x1234 es palabra 0x34 dentro de página 0x12 cuya primera (menor) dirección es 0x1200

30. 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

Re:ver T4 tr.15-19: $T=1\text{ns}$, idealmente $T_c=0.25\text{ns}$, $k+n-1=9+4-1=12\text{ciclos}=3\text{ns}$
