

COGNOMS:

[illegible]

NOM:

[illegible]

**IMPORTANTE leer atentamente antes de empezar el examen:** Escriba los apellidos y el nombre antes de empezar el examen. Escriba un solo carácter por recuadro, en mayúsculas y lo más claramente posible. Es importante que no haya tachones ni borrones y que cada carácter quede enmarcado dentro de su recuadro sin llegar a tocar los bordes. Use un único cuadro en blanco para separar los apellidos y nombres compuestos si es el caso. No escriba fuera de los recuadros.

### Problema 1. (2,5 puntos)

Un programa P ejecuta  $5 \times 10^9$  instrucciones dinámicas y realiza  $3 \times 10^9$  operaciones de punto flotante. El programa P tarda en ejecutarse  $8 \times 10^9$  ciclos en un procesador C que funciona a una frecuencia de 4 GHz.

a) **Calcula** el CPI y el tiempo de ejecución en segundos (T<sub>exec</sub>) del programa P.

CPI =  $8 \times 10^9$  ciclos /  $5 \times 10^9$  instrucciones = **1,6 ciclos / instrucción**

$$\text{Texe} = \text{Ciclos} / F = 8 \times 10^9 \text{ ciclos} / 4 \text{GHz} = \mathbf{2 \text{ seg}}$$

El procesador C tiene una corriente de fuga de 10 A, se alimenta a un voltaje de 2 V y tiene una carga capacitiva equivalente de 5 nF. El consumo debido a cortocircuito es despreciable.

b) **Calcula** la energía consumida al ejecutar el programa P en el procesador C.

$$P = P_{fuga} + P_{conmut} = I \cdot V + C \cdot V^2 \cdot F = 10 \text{ A} \cdot 2 \text{ V} + 5 \text{ nF} \cdot (2 \text{ V})^2 \cdot 4 \text{ GHz} = 100 \text{ W}$$

$$E = P \cdot t = 100\text{W} \cdot 2\text{ s} = \mathbf{200\text{ Joules}}$$

El programa P tiene un 80% del código que es perfectamente paralelizable.

c) **Calcula** el número (N) de procesadores idénticos a C que son necesarios para conseguir un speed-up de 4 en el programa P.

Ley de amdahl:

$$4 = 1/(0,2 + 0,8/N) \rightarrow N = 16$$

d) **Calcula** los MFLOPS de P en dicho sistema paralelo (N procesadores C).

$$T_{\text{exeP}} = T_{\text{exe}}/4 = 0,5s$$

$$\text{MFLOPS} = \text{OpsPF} / \text{TexeP} * 10^6 = 3 \times 10^9 \text{ opsPF} / 0,5 * 10^6 = \mathbf{6000 \text{ MFLOPS}}$$

En un sistema paralelo con 25 procesadores idénticos a C, cada procesador tiene un MTTF de 10 millones de horas. El resto del sistema (sin contar las CPUs) tiene un MTTF de 100.000 horas.

e) **Calcula** el tiempo medio hasta fallo (MTTF) del sistema completo.

$$\text{MTTF} = 1/(25/10.000.000 + 1/100.000) = 80.000 \text{ horas}$$





COGNOMS:

NOM:

**Problema 4. (2.5 puntos)**

Cuando se va a ejecutar la siguiente instrucción escrita en ensamblador del x86:

```
movl $337, -40(%ebp, %esi, 4)
```

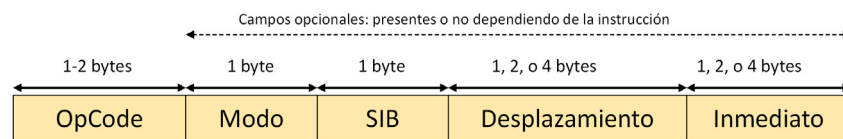
El contenido de los registros del procesador (en hexadecimal) es:

```

eax: 0x00000005    ebx: 0x00000004    ecx: 0x00000001    edx: 0x00000002
esp: 0x80800f30    ebp: 0x80800f48    esi: 0x00000002    edi: 0x00001001
eip: 0x808000fe    eflags: 0x286

```

Sabemos que la codificación de instrucciones sigue el siguiente esquema tal y como se explica en clase:



Además sabemos que el procesador tiene una cache de datos de 32Kbytes (copy back+write allocate), y otra de instrucciones de 16Kbytes, ambas con líneas de 32 bytes y asociatividad 2. Sus páginas son de 4Kbytes y dispone de un TLB de 4 entradas. Asumimos que tanto el TLB como las caches están vacíos.

a) ¿**Cuántos** bytes ocuparía la instrucción anterior si sabemos que el OpCode de movl solo ocupa 1 byte?

1 opcode + 1 Modo + 1 SIB + 1 Desplazamiento + 2 Inmediato = 6 bytes

b) ¿**Cuántos** bloques de memoria se leen para ejecutar el movl? **Escribe** la dirección de dichos bloques.

2 instrucciones (0x808000e0 y 0x80800100) + 1 datos (0x80800f20)

c) ¿**Cuántos** bloques de memoria se escriben para ejecutar el movl? **Escribe** la dirección de dichos bloques.

Ninguno, es copy back + write allocate

d) ¿**Cuántos** fallos de TLB tendremos al ejecutar la instrucción anterior? **Indica** a que páginas es el fallo

1 fallo de TLB, instrucciones (0x808000) y datos están en la misma página

Suponiendo el siguiente código:

```

leal N-1(), %esi          <- Esta instrucción ocupa 2 bytes
bucle: movl $337, -40(%ebp, %esi, 4) <- Instrucción a evaluar
      decl %esi
      jns bucle            <- El bucle hace N iteraciones

```

e) ¿**Cuántos** fallos de cache provoca el movl en función del valor de N? (suponemos de nuevo que las caches están vacías y los registros tienen valor indicado anteriormente antes de ejecutar el código).

1 + N/8 + 1 si N no es múltiplo de 8: 1 para las instrucciones + N/8 para los datos pero aumenta en 1 fallo extra si N no es múltiplo de 8 por alineamiento.