COGNOMS:															
NOM:									NI/N	IIE:					

IMPORTANTE leer atentamente antes de empezar el examen: Escriba los apellidos, el nombre y el DNI/NIE 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, todo lo que haya fuera de ellos es ignorado. La identificación del alumno se realiza de forma automática, no seguir correctamente estas instrucciones puede comportar no tener nota.

Problema 1. (3 puntos)

Disponemos de un computador con un procesador a 2GHz. Un programa P ejecuta 1000 millones de instrucciones que se distribuyen de la siguiente manera:

	punto flotante	enteras	memoria
Numero de instrucciones	200 millones	400 millones	400 millones
CPI medio	5	2	8

Cada instrucción de punto flotante es una instrucción SIMD que realiza 4 operaciones de coma flotante.

a) Calcula el tiempo de ejecución del programa P.

```
Ciclos = 200x10^6 x 5 + 400x10^6 x 2 + 400x10^6 x 8 = 5x10^9 ciclos

Texe=5x10^9 / 2x10^9=2,5 s
```

b) Calcula el CPI del programa P.

CPI=5x10^9 ciclos / 1x10^9 i = 5 c/i

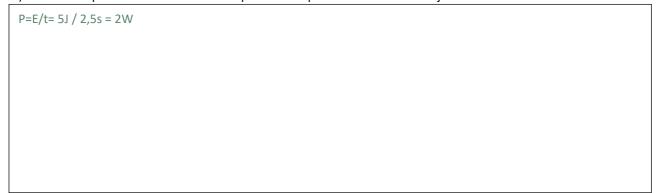
c) ¿Es posible conseguir que el programa vaya el doble de rápido solo acelerando las instrucciones de memoria? Si la respuesta es sí **calcula** el CPI medio que tendrían que tener dichas instrucciones para conseguirlo.

SI Ciclos mem= 2,5x10^9 - 1,8x10^9 = 7x10^8 ciclos CPI mem= 700x10^6 ciclos / 400x10^6 ins = 1,75 c/i

1 December 2020 10:48 am

La carga de una pila o batería se mide en Amperios x hora (Ah), que indica la cantidad de corriente que puede entregar de forma continua durante una hora antes de quedar descargada totalmente. Nuestro procesador se alimenta con una batería de 8Ah a 1,25 V. El procesador consume 5 J durante la ejecución del programa P.

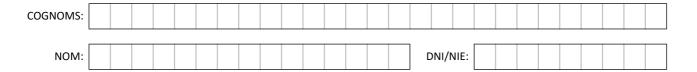
d)	Calcula la potencia me	edia consumida por	nuestro procesador	durante la eiecución de P	



e) Calcula la duración de la batería si el procesador ejecuta el programa P de forma continua?



1 December 2020 10:48 am 2/6



Problema 2. (3.5 puntos)

Considera el siguiente código escrito en C, suponiendo que las constantes A, B y C han sido declaradas previamente en un #define.

```
typedef struct {
  int t2[B];
  int y;
  int t3[C];
} sa;

typedef struct {
  sa tabla[A];
  int x;
} sb;

void examen(sb *p1, int val)
{
  int idc;
  idc = p1->x;
  p1->tabla[idc].y = val;
}
```

a) **Dibuja** como quedarían almacenadas en memoria las estructuras sa y sb, indicando claramente los deplazamientos respecto del inicio y el tamaño de todos los campos. Escribir esos valores en función de A, B y C.

```
----- <---- 0
                                              ----- <---- 0
  | t2[0] |
                                             | tabla[0] |
                                                ----- <---- tamaño de sa (= 4*(B+1+C))
  | ... |
                                              | ... |
  _____
  | t2[B-1] |
                                             | tabla[A-1] |
                                              ----- <---- A*tamaño de sa (= 4*A*(B+1+C))
  ----- <---- 4*B
     у |
  ----- <---- 4*(B+1)
                                             ----- <---- (A*tamaño de sa) + 4
  | t3[0]
          ----- <---- 4*(B+2)
                                            Tamaño de sb = (A*tamaño de sa) + 4
  | ... |
                                            =4*[A*(B+1+C)+1]
                                            Alineamiento a 4 bytes
 | t3[C-1] |
  ----- <---- 4*(B+1+C)
tamaño de sa = 4*(B+1+C)
Alineamiento a 4 bytes
```

1 December 2020 10:48 am 3/6

b) **Dibuja** el Bloque de Activación de la función examen, indicando claramente los desplazamientos relativos al EBP necesarios para acceder a los parámetros y las variables locales.

c) **Determina** los valores de las constates A, B y C sabiendo que el código en ensamblador del x86 de las dos instrucciones que hay dentro de la función examen es el siguiente. Suponer que en el registro %edi se encuentra 8(%ebp) y en %esi se encuentra val.

```
examen:
```

```
movl 168(%edi), %eax
leal (%eax, %eax, 2), %eax
movl %esi, 12(%edi, %eax, 8)
```

```
A = 7
B = 3
C = 2
```

1 December 2020 10:48 am 4/6

COGNOMS:															
NOM:								[NI/N	IIE:					

Problema 3. (3.5 puntos)

Un procesador consta de una memoria cache con escritura inmediata (write through) y sin carga en caso de fallo de escritura. Mediante contadores hardware se han obtenido las siguientes medidas para un conjunto de programas de referencia:

- porcentaje de escrituras: 20%
- porcentaje de bloques modificados al substituir una línea: 33.33%
- tasa de aciertos 0.9

El tiempo de acceso a memoria cache es de 5 ns y el tiempo de memoria principal para escribir una palabra es de 70 ns. Para leer o escribir un bloque en la memoria principal se emplean 90 ns.

a) Calcula el tiempo invertido en realizar 1000 accesos a memoria.

```
1000 accesos 200 escrituras hit+miss 70ns 14000 ns

800 lecturas 720 hits 5ns 3600 ns

80 miss 100 ns 8000 ns

25600 ns
```

La empresa A.C.M.E está diseñando un procesador que sabemos que genera 1.3 referencias a memoria por instrucción de las cuales 0.3 son de datos. Sus diseñadores se han dado cuenta de que en el mismo chip queda espacio suficiente para poner un poco de cache. Después de hacer algunos cálculos han optado por poner dos caches diferenciadas con la siguientes características:

- Dos caches separadas, una de instrucciones de 4Kb y una de datos de 8Kb. El CPI ideal es de 1.2 ciclos/instrucción ya que podemos buscar datos e instrucciones simultáneamente.
- Tc (tiempo de ciclo): 10ns
- Tsa (tiempo de servicio en caso de acierto): 1ciclo.
- Tsf (tiempo de servicio en caso de fallo): 10 ciclos.

La tasa de fallos para la cache de instrucciones es del 8.6% y para la cache de datos es del 6.8%.

b) Calcula el tiempo medio de ejecución de 1 instrucción

```
CPI Real=CPI Ideal + 1 * 9 * 0.086 + 0.3 * 9 * 0.068 = 2,1576

Texe= CPI Real * Tc = 21,576 ns
```

1 December 2020 10:48 am 5/6

Dado el siguiente código escrito en ensamblador x86:

```
movl $0, %ebx
movl $0, %esi
for: cmpl $2048*1000, %esi
    jge end
    movw (%ebx, %esi, 2), %ax
    addw %ax, 16*1024(%ebx, %esi, 2)
    movw %ax, 32*1024(%ebx, %esi, 2)
    addl $2048, %esi
    jmp for
end:
```

La memoria utiliza páginas de tamaño 8KB y que utilizamos un TLB de 4 entradas (reemplazo LRU).

c) Calcula el porcentaje de aciertos de TLB en todo el bucle.

Las páginas tienen tamaño 8KB y cada iteración del bucle incrementa el acceso en 4KB así que cada 2 iteraciones se accede a una página distinta. Por otro lado en la iteración 0, la primera instrucción accede a la página 0, la segunda a la página 2 y la tercera a la página 4, en la iteración 2 acceden a las páginas 1, 3 y 5 respectivamente de forma que no se pueden reutilizar accesos. Por lo tanto tendremos 3 fallos cada 2 iteraciones u 8 accesos.

forma que no se pueden reutilizar accesos. Por lo tanto tendremos 3 fallos cada 2 iteraciones u 8 accesos.

Aciertos = 5/8 => 62,5%

1 December 2020 10:48 am 6/6