

COGNOMS: NOM:

1er Control Arquitectura de Computadors

Curs 2015-2016 Q1

- Temps: 13:30 a 15:00
- Poseu clarament amb LLETRES MAJÚSCULES a cada full els cognoms i el nom

Problema 1. (4 puntos)

Un programa (P) es ejecutado en una CPU (C). La siguiente tabla muestra la distribución de instrucciones dinámicas para el programa (P) junto con el CPI medio de cada tipo de instrucción para la CPU (C).

	punto flotante	enteras	memoria
% instrucciones	40%	35%	25%
CPI	4,0	2,0	6,8

- a) **Calcula** el CPI del programa (P).

Hemos podido medir que el rendimiento en MIPS del programa (P) es de 600 MIPS.

- b) **Calcula** la frecuencia de la CPU (C).

Se ha decidido que una rutina (R), que representa el 80% del tiempo de (P), sea programada en ensamblador.

- c) **Calcula** la ganancia en tiempo de ejecución (speedup) que se debería obtener para la rutina (R) respecto al compilador para que el programa (P) se ejecute 2,5 veces más rápido.

La CPU (C) tiene una capacidad efectiva equivalente de 12,5 nF (nanofaradios), una corriente de fugas de 10 A y funciona a un voltaje de 1,2 V.

- d) **Calcula** la potencia debida a fugas, la potencia debida a conmutación y la potencia total disipada por la CPU (C).

Este computador está formado por los componentes mostrados en la tabla siguiente. La tabla también muestra el número de componentes de cada tipo y el tiempo medio hasta fallo (MTTF) de cada componente.

Componente	Fuente alimentación	CPU	Ventilador CPU	Placa base	DIMMs	Discos duros	Tarjetas graficas
Nº	1	1	1	1	4	2	2
MTTF (horas)	100.000	1.000.000	100.000	200.000	1.000.000	125.000	500.000

El tiempo medio para reemplazar un componente que ha fallado (*mean time to repair*) es de 5 horas y la probabilidad de fallo sigue una distribución exponencial.

- e) **Calcula** el tiempo medio hasta fallos del hardware (MTTF), el tiempo medio entre fallos (MTBF) y la disponibilidad del sistema.

COGNOMS: NOM:

1er Control Arquitectura de Computadors

Curs 2015-2016 Q1

Problema 2. (3 puntos)

Dado el siguiente código escrito en C:

```
int exaFacil(int i, int v[10]) {  
    int tmp;  
    tmp = -1;  
    if (i < 10)  
        tmp = v[i];  
    return tmp;  
}
```

- a) **Dibuja** el bloque de activación de la rutina exaFacil, indicando claramente los desplazamientos respecto a ebp y el tamaño de todos los campos.

- b) **Traduce** a ensamblador del x86 la rutina exaFacil.

COGNOMS: NOM:

1er Control Arquitectura de Computadors

Curs 2015-2016 Q1

Problema 3. (3 puntos)

Dado el siguiente código escrito en C:

```
typedef struct {  
    char a;  
    short int b[2001];  
    char c;  
} X;  
  
X S;
```

- a) **Dibuja** la estructura S identificando claramente el tamaño de la estructura, el tamaño de cada elemento y el offset de cada elemento respecto al inicio de la estructura.

- b) **Escribe** una instrucción que permita almacenar S.c en la parte baja del registro **%eax** suponiendo que la dirección de la estructura está almacenada en el registro **%ebx**.

- c) **Escribe** una instrucción que permita almacenar S.b[i] en la parte baja del registro **%eax** suponiendo que la dirección de la estructura está almacenada en el registro **%ebx** y que la variable i está almacenada en el registro **%ecx**. **Escribe** claramente la expresión aritmética que has usado para el cálculo de la dirección.