

# Aspectos Avanzados de Arquitectura de Computadoras y Taller de Arquitectura de Computadoras

Prueba escrita correspondiente al curso 2012

- Indique su nombre completo y número de cédula en cada hoja.
- Numere todas las hojas e indique el total de hojas en la primera. Escriba las hojas de un solo lado.
- Solo se responderán dudas de letras. No se responderán preguntas en los últimos 30 minutos de la prueba.
- La prueba es individual y sin material. Duración de la prueba : 2 horas
- Todas las preguntas tienen el mismo puntaje y el mínimo de aprobación es un un 60%.

## Pregunta 1

Explique que recursos de hardware son necesarios para explotar el paralelismo a nivel de instrucción. Cuál es la función de la etapa de commit en superescalares?

## Pregunta 2

Explique por qué MIPS y CPI pueden no ser buenos indicativos del rendimiento de una CPU. Ilustre con ejemplos.

## Ejercicio 1

Se dispone de una lista enlazada en memoria, con la siguiente definición:

```
struct Item {
    int data;
    Item *next;
}
```

La lista vacía se representa mediante un puntero con valor NULL.

La siguiente función recursiva calcula la suma de los elementos de una lista:

```
int suma_lista(*Item) {
    if (Item == NULL) return 0;
    int data = Item->data;
    Item *next = Item->next;
    return data + suma_lista(next);
}
```

Compilar en ensamblador de Intel 8086. El parámetro Item se pasa en el registro BX como un desplazamiento dentro del segmento apuntado por DS y el resultado se devuelve en AX.

## **Solución Ejercicio 1**

```
proc suma_lista
    push cx                ;guardo registro auxiliar
    cmp bx,NULL           ;si el puntero es NULL
    je paso_base          ;salto al paso base

    ;si no es NULL
    mov cx, [bx]          ;guardo el valor del item actual
    mov bx, [bx+2]        ;actualizo el puntero al siguiente
    call suma_lista        ;llamada recursiva
    add ax,cx              ;sumo valor actual + retornado por suma_lista
    jmp fin                ;salto al final

paso_base:
    mov ax, 0              ;paso base es retornar cero
fin:
    pop cx                 ;dejo el stack como estaba originalmente
    ret
endp
```

## **Ejercicio 2**

Un computador contiene una memoria principal de 32K palabras de 16 bits.

Tiene también una caché de 4K palabras dividida en conjuntos de 4 líneas con 64 palabras por línea. Suponga que la caché está inicialmente vacía. El procesador capta palabras de las posiciones 0,1,2,.....,4351 en ese orden. Entonces repite la secuencia de captación 9 veces más.

La caché es 10 veces más rápida que la memoria principal.

Estime la mejora resultante por el uso de la caché. Suponga una política LRU para la sustitución de bloques.

## **Solución Ejercicio 2**

La memoria principal consiste de 512 bloques de 64 palabras.

La cache tiene 16 conjuntos, cada conjunto de 4 vías y cada vía de 64 palabras.

Los lugares 0 al 4351 en la memoria principal ocupan los bloques 0 al 67.

En el primer fetch, se leen del bloque 0 al 15 y se colocan en los conjuntos 0 al 15 de la cache, luego se leen del 16 al 31 y se colocan también en los conjuntos 0 al 15, los bloques del 31 al 47 se colocan también del 0 al 15, lo mismo para los bloques del 48 al 63. Finalmente los bloques del 64

al 67 del 0 al 3.

Como cada conjunto tiene 4 vías no hay reemplazos hasta el bloque 63. Los últimos 4 bloques implican un reemplazo. En cada pasada, los reemplazos serán necesarios en los conjuntos 0 al 3 pero todos los bloques del 4 al 15 se mantienen sin cambios. Por lo tanto, en cada pasada, 48 bloques se mantienen igual y 20 deben ser leídos.

Sea  $T$  el tiempo para leer un bloque (64 palabras) de la cache. Luego  $10T$  es el tiempo para leer un bloque de la memoria principal. Si una palabra no esta en la cache, se necesita leer de la memoria y clocar en la cache. Entonces el tiempo para leer un bloque de 64 palabras de la cache si este no esta en la misma es de  $11T$ .

Entonces podemos escribir la mejora de la siguiente manera.

Sin cache: Tiempo de lectura = (10 pasadas) x (68 bloques/pasada) x (10T/bloque) = 6800T

Con cache: Tiempo de lectura = tiempo primer pasada + tiempo restante pasadas =  
 = (68 bloques/pasada) x (11T/bloque) + 9 x (48 bloques/pasada) x (1T/bloque) + 9 x (20 bloques/pasada) x (11T/bloque) = 3160T

$$\text{Mejora} = \frac{6800T}{3160T} = 2.15$$

Otra solución (pensando los accesos de a palabra):

El tiempo de captar de memoria una palabra o un bloque es el mismo ( $10T$ ).

Sin cache: Tiempo de lectura = (10 pasadas) x (4352 palabras/pasada) x (10T/palabra) = 435200T

Con cache: Tiempo de lectura = tiempo primer pasada + tiempo restante pasadas =  
 =[copiar bloques de memora a cache(misses) + leer palabras (de esos bloques) de cache] + [leer palabras de cache(hits) + copiar bloques de memoria a cache(misses) + leer palabras (de esos bloques) de cache] =

= [(68 bloques/pasada) x (10T/bloque) + 4352 x (1T/palabra)] + [9 x (3072 palabras/pasada) x (1T/palabra) + 9 x (20 bloques/pasada) x (10T/bloque) + 9 x (1280 palabras/pasada) ] x (1T/palabra) = 46000

$$\text{Mejora} = \frac{435200T}{46000T} = 9.46$$