

UNIVERSIDAD DE COSTA RICA ESCUELA DE INGENIERÍA ELÉCTRICA

EIE

Escuela de Ingeniería Eléctrica

Ejercicios de Práctica

IE-0321 Estructuras de Computadores Digitales I Profesor: Jose Ariel Fallas Pizarro

1) Escriba en lenguaje ensamblador de Mips el siguiente código que está en alto nivel.

```
for (i=1; i<129;i=i*2)
{
     A[i]=A[i-1]+A[i+1];
}</pre>
```

suponga que i se mapea en \$t0 y A en \$s0.

2) Escriba en lenguaje ensamblador de Mips el siguiente código que está en alto nivel.

```
i=0;
while (A[2*i]!=0)
{
    A[i]=A[2*i-1]+A[2*i+1];
    i++;
}
```

suponga que i se mapea en \$t0 y A en \$s0.

3) Escriba en lenguaje ensamblador de Mips el siguiente código que está en alto nivel.

suponga que i se mapea en \$t0 y A en \$s0.

4) Para el procesador MIPS segmentado de 5 etapas visto en clase. Si no hay anticipación o detección de riesgos, en el siguiente código inserte instrucciones nop para garantizar una ejecución correcta.

```
add $t5,$t2,$t1
lw $t3,4($t5)
lw $t2,0($t2)
or $t3,$t5,$t3
sw $t3,0($t5)
```

Para el procesador MIPS segmentado de 5 etapas visto en clase. Si no hay anticipación o detección de riesgos, en el siguiente código inserte instrucciones nop para garantizar una ejecución correcta. (Asuma el peor caso de ejecución)

```
add $t5,$t2,$t1
lw $t3,4($t5)
beq $t3,0,jump
lw $t2,0($t2)
jump :
add $t3,$t2,$t3
sw $t3,0($t5)
```

- 6) Escriba, en lenguaje ensamblador de mips, una función que reciba por medio de \$a0 la dirección de un array A y por \$a1 N (el número de palabras de este array). Este array corresponde a números enteros sin signo. La función debe devolver por \$v0 el número mayor del array y por \$v1 el número menor del array.
- 7) Escriba, en lenguaje ensamblador de mips, una función que reciba por medio de \$a0 la dirección de un array A y por \$a1 N (el número de palabras de este array). Este array corresponde a números enteros sin signo. La función debe devolver por \$v0 un uno si hay dos números iguales consecutivos en el array y un cero en caso contrario.

- 8) Supongamos un sistema con un procesador que lanza direcciones de 16 bits. Dispone de una cache de 64 bytes, organizada en bloques de 32 bytes. En la siguiente tabla aparece una secuencia de direcciones de datos (en hexadecimal). Supondremos que se repite dos veces la misma secuencia. Indique sobre la tabla, para cada acceso, si produce un fallo o un acierto suponiendo mapeo directo y cache inicialmente vacía.
 - a. Calcule la cantidad de bits que se utilizan para offset
 - b. Calcule la cantidad de bits que se utilizan como index
 - c. Calcule la cantidad de bits que se utilizan como TAG
 - d. Calcule la tasa de aciertos

Dir	0x0212	0x0351	0x0215	0x025F	0x023C	0x0350	0x0240	0x14A1	0x514C	0x14A4	0x021A
It 1											
It 2											

- 9) Supongamos un sistema con un procesador que lanza direcciones de 32 bits. Dispone de una cache de 1024 bytes, organizada en bloques de 128 bytes. En la siguiente tabla aparece una secuencia de direcciones de datos (en hexadecimal). Supondremos que se repite dos veces la misma secuencia. Indique sobre la tabla, para cada acceso, si produce un fallo o un acierto suponiendo mapeo asociativo por conjuntos en 2 vias y con política de reemplazo LRU.
 - e. Calcule la cantidad de bits que se utilizan para offset
 - f. Calcule la cantidad de bits que se utilizan como index
 - g. Calcule la cantidad de bits que se utilizan como TAG
 - h. Calcule la tasa de aciertos

Dir	0x 053407A1	0x 05540903	0x A7934021	0x 025FF942	0x 025FC92A	0x A792752C	0x 0B5A3B0F	0x 14A19A01	0x 05540B13	0x 14A49241	0x 14A4C241
lt 1											
It 2											