



Tema 4: PROBLEMAS

Segmentación *-pipelining-*

Arquitectura de Computadores
Grado en Ingeniería Informática

Tema 4: PROBLEMAS

Problema 1

Sea la siguiente secuencia de instrucciones:

ADD \$8, \$9, \$10

LW \$11, 4(\$8)

LW \$12, 8(\$8)

Contenido de ciertos registros y posiciones de memoria:

\$8 = 0

\$9 = 20

\$10 = 12

MEM[36] = 200

MEM[40] = 100

- 1.- Indica las dependencias que existen entre las instrucciones.
- 2.- Supongamos que esta secuencia de instrucciones se ejecuta en un procesador MIPS-32 segmentado CON unidad de detección de riesgos (bloqueos) y unidad de anticipación . Representar en un diagrama qué sucede en cada ciclo de ejecución.

Tema 4: PROBLEMAS

Problema 2

Sea la siguiente secuencia de instrucciones:

```
LW $8, 4($9)
ADD $11, $8, $10
SUB $12, $12, $12
```

Contenido de ciertos registros y posiciones de memoria:

```
$8 = 3
$9 = 16
$10 = 2
$12 = 1
MEM[20] = 200
```

1.- Indica las dependencias que existen.

2.- Supongamos que esta secuencia de instrucciones se ejecuta en un procesador MIPS-32 segmentado CON unidad de detección de riesgos (bloqueos) y unidad de anticipación . Representar en un diagrama qué sucede en cada ciclo de ejecución.

Tema 4: PROBLEMAS

Problema 3

Sea la siguiente secuencia de instrucciones:

LW \$1, 40(\$7)

SW \$6, 8(\$1)

ADD \$9, \$2, \$2

1. Indica las dependencias que existen.
2. Supongamos que esta secuencia de instrucciones se ejecuta en un procesador MIPS-32 segmentado CON unidad de detección de riesgos (bloqueos) y unidad de anticipación. Representar en un diagrama qué sucede en cada ciclo de ejecución.
3. Si fuese el compilador el encargado de resolver los riesgos, ¿qué reordenación de código haría? ¿Tendría el compilador que insertar alguna instrucción NOP?

Tema 4: PROBLEMAS

Problema 4

Sea la siguiente secuencia de instrucciones:

LW \$5, -16(\$5)

SW \$5, -16(\$5)

ADD \$5, \$5, \$5

1. Indica las dependencias que existen.
2. Supongamos que esta secuencia de instrucciones se ejecuta en un procesador MIPS-32 segmentado CON unidad de detección de riesgos (bloqueos) pero SIN unidad de anticipación. Representar en un diagrama qué sucede en cada ciclo de ejecución.
3. ¿Qué ocurre si al procesador se le añade una unidad de anticipación? ¿Se resuelven los riesgos?
4. Si fuese el compilador el encargado de resolver los riesgos, ¿qué reordenación del código haría? ¿Tendría el compilador que insertar alguna instrucción NOP?