

# Arquitectura y Tecnología de Computadores

## Arquitecturas Paralelas. Curso 10/11

### Lección 5 - Ejercicio 1

- 1) ¿Cuál será la ganancia de velocidad obtenida al segmentar un procesador con 20 etapas, si ejecuta un programa de 50 instrucciones?
- 2) ¿y si ejecuta un programa de 500 instrucciones?

### Lección 5 - Ejercicio 2

Considerar una máquina no segmentada con cinco pasos de ejecución cuyas duraciones son 50 ns, 50 ns, 60 ns, 50 ns y 50 ns. Suponer que, debido al tiempo de preparación y sesgo de reloj, segmentar la máquina añade 5 ns de gasto a cada etapa de ejecución. Ignorando cualquier impacto de latencia, ¿qué velocidad se ganará con la segmentación?

### Lección 5 - Ejercicio 3

Suponer que las referencias a datos constituyen el 30 por ciento de la mezcla de instrucciones de un programa y que el CPI de la máquina segmentada, ignorando los riesgos estructurales, es 1,2. Sin considerar ninguna otra pérdida de rendimiento, ¿cuántas veces es más rápida la máquina sin los riesgos estructurales de memoria, frente a la máquina con los riesgos?

### Lección 5 - Ejercicio 4

Muchas máquinas recientes no tienen unidades de punto flotante totalmente segmentadas. Por ejemplo, supongamos que tenemos un procesador con una latencia de 5 ciclos de reloj para multiplicar en punto flotante, pero sin segmentación. ¿Tendrá este riesgo estructural un impacto grande o pequeño sobre el rendimiento durante la ejecución del programa *SPICE*, si la multiplicación flotante tiene una frecuencia del 5% en dicho programa? (considerar las multiplicaciones distribuidas en el código de forma uniforme)

### Lección 5 - Ejercicio 5

Suponer que el 20% de las instrucciones de una cierta carga de trabajo para el procesador segmentado utilizado como ejemplo son instrucciones de carga, y que la mitad de las veces la instrucción que sigue a una instrucción de carga depende de la carga. Si este problema da lugar a un retardo de 1 ciclo, ¿cuántas veces es más rápida la máquina ideal segmentada si se compara con una segmentación más realista?

### **Lección 5 - Ejercicio 6**

Proponer una secuencia de código convencional para el procesador de ejemplo, que realice las operaciones de alto nivel:

$$a = b + c$$

$$d = e - f$$

Estimar el número de ciclos perdidos y proponer una secuencia de código alternativa, planificada por el compilador, que evite dicha pérdida, ¿que inconveniente aparece?

### **Lección 5 - Ejercicio 7**

Se tiene un procesador segmentado de 5 etapas con un CPI ideal de 1 y hueco de retardo de salto de 2 ciclos. Además, la dirección de salto efectivo no se conoce hasta el final del primer ciclo del hueco. ¿Cuál será el CPI real al ejecutar un programa de 80 instrucciones, de las cuales 8 dan lugar a salto efectivo y otras 8 a salto NO efectivo, si se sigue la estrategia de predicción de “salto efectivo”?

### **Lección 5 - Ejercicio 8**

A partir de las gráficas y datos relativos a riesgos de control para la arquitectura del procesador segmentado visto en clase, y considerando un CPI ideal sin riesgos de 1, se pide:

- 1) ¿Cual es la frecuencia de instrucciones de control?
- 2) ¿Qué porcentaje de las instrucciones de control cambia realmente el contador de programa?
- 3) ¿Qué porcentaje medio de saltos retardados hace realmente trabajo útil?
- 3) Rellenar la tabla siguiente relativa a los costes asociados a varios esquemas de planificación de salto:

<b>Esquema de planificación</b>	<b>Penalización de salto</b>	<b>CPI efectivo o real</b>	<b>Aceleración de la segmentación (Ganancia)</b>	<b>Aceleración respecto a la estrategia de detención</b>
1- Detención	2			
2- Predicción “salto efectivo”	1			
3- Predicción “salto no efectivo”	1			
4- Salto retardado	1			