

Instituto Tecnológico de Costa Rica Área Académica de Ingeniería en Computadores Programa de Licenciatura en Ingeniería en Computadores Curso: CE-4301 Arquitectura de Computadores I Profesor: Lic. Ing. Fabián Zamora Ramírez Semestre: II, 2017	Tarea No.2 Fecha de entrega: 20 de Setiembre, 2017 Tema: Práctica para examen I Puntos totales: 100 Puntos obtenidos: _____
---	--

Nombre del estudiante: _____

Carné del estudiante: _____

INSTRUCCIONES GENERALES.

- Esta evaluación debe ser entregada individualmente.
- Presente este enunciado como parte de su solución.
- Responda de forma clara y ordenada.
- La entrega de esta tarea es impresa o escrita.

EJERCICIOS

Conteste las siguientes preguntas de manera adecuada según se le solicita. Realice el planteo del problema y todos los procedimientos necesarios para llegar a la solución correcta.

TEORIA

1. Mencione las categorías en las que dividió Flynn los procesadores. Presente un ejemplo de al menos dos de ellas.

AMDAHL

2. Asuma que se hace un cambio a un computador que mejora cierto modo de ejecución (A) en un factor de 10. Este modo A es utilizado el 50% del tiempo (este 50% es del tiempo cuando se utiliza el modo ya mejorado). Recuerde que la ley de Amdahl depende de la fracción del tiempo de ejecución SIN MEJORAR que puede ser mejorado. De forma que no se puede utilizar este 50% para calcular el speedup utilizando la ley de Amdahl. Sabiendo que:

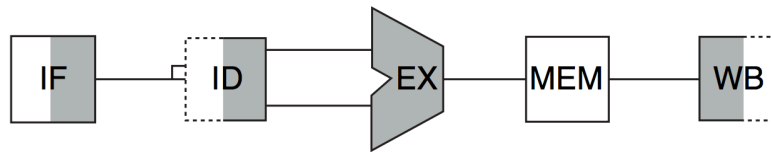
$$\text{Speedup}_{\text{overall}} = \frac{\text{Execution time}_{\text{old}}}{\text{Execution time}_{\text{new}}} = \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}}$$

- a) ¿Cuál es el speedup obtenido utilizando el modo mejorado?
 - b) ¿Qué porcentaje del tiempo de ejecución original se ha mejorado?
3. Al hacer cambios para optimizar parte de un computador, en ocasiones, al mejorar una característica se empeora otra. Por ejemplo, al mejorar una unidad de punto flotante compleja, el área agregada puede influir negativamente en el tiempo de acceso a la memoria cache de primer nivel (L1), agregando un ciclo extra. La ley de Amdahl, como tal, no toma en cuenta estas situaciones.
- a. Si la nueva unidad de punto flotante mejora las operaciones en 2x, y dichas operaciones toman 20 % del tiempo de ejecución del sistema original ¿Cuál es la mejora general del sistema?
 - b. Ahora asuma que la nueva unidad de punto flotante (descrita en a.) empeora los accesos a memoria cache L1, resultando en un empeoramiento de los mismos de 1.5x (mejora de 2/3x). Si los accesos a caché L1 consumen 10 % del tiempo de ejecución del sistema original ¿Cuál es ahora la mejora general en el sistema?
4. Su compañía acaba de comprar un nuevo procesador Intel Core i5, de doble núcleo, y se le ha asignado a usted optimizar el software para ese procesador. Usted ejecutará dos aplicaciones en el procesador, pero los recursos requeridos no son iguales. La primera aplicación requiere 80% de los recursos y la otra el 20% restante. Asuma que cuando

paraleliza una porción del programa, la mejora de esa porción es de 2.

- c. Dado que el 40% de la primera aplicación es paralelizable ¿Qué mejora obtendría en el sistema si solo esta aplicación se ejecuta?
 - d. Dado que el 99% de la segunda aplicación es paralelizable ¿Qué mejora obtendría en el sistema si solo esta aplicación se ejecuta?
 - e. Dado que el 99% de la segunda aplicación es paralelizable ¿Qué mejora observaría el *sistema general* (ambas aplicaciones se ejecutan)?
5. Al paralelizar una aplicación, la mejora ideal es equivalente al número de procesadores. Esto es limitado por dos aspectos: porcentaje de la aplicación paralelizable y costo de comunicación. La ley de Amdahl toma en cuenta el primer factor, pero no el segundo.
- f. ¿Cuál es la mejora con 8 procesadores si el 80% de la aplicación es paralelizable, ignorando la comunicación?
 - g. ¿Cuál es la mejora con 8 procesadores si, para cada procesador, el *overhead* de comunicación es de 0,5% del tiempo de ejecución original?
 - h. ¿Cuál es la mejora con N procesadores si, por cada vez que se dobla la cantidad de procesadores, el *overhead* de comunicación aumenta un 0,5% del tiempo de ejecución original?
 - i. Determine una ecuación para la cantidad de procesadores, con la máxima mejora, en una aplicación con P% de la aplicación paralelizable y, por cada vez que se dobla la cantidad de procesadores, el *overhead* de comunicación aumenta un 0.5% del tiempo de ejecución original.

SEGMENTACION



6. Para las siguientes secuencias de instrucciones:

	Instruction sequence
a.	<code>lw \$1,40(\$2)</code> <code>add \$2,\$3,\$3</code> <code>add \$1,\$1,\$2</code> <code>sw \$1,20(\$2)</code>
b.	<code>add \$1,\$2,\$3</code> <code>sw \$2,0(\$1)</code> <code>lw \$1,4(\$2)</code> <code>add \$2,\$2,\$1</code>

- Encuentre todas las dependencias de datos.
- Encuentre todos los riesgos para un pipeline de 5 etapas si no se utiliza adelantamiento.
- Encuentre todos los riesgos para un pipeline de 5 etapas utilizando adelantamiento.
- Para reducir el tiempo del ciclo de reloj, se considera separar la etapa de MEM en 2 etapas. Repita los puntos b y c para este pipeline con 6 etapas.

Asuma que antes de ejecutar las instrucciones anteriores se tienen los siguientes valores para los registros \$0 a \$3:

	\$0	\$1	\$2	\$3
a.	0	1	31	1000
b.	0	-2	63	2500

- ¿Cuáles son los valores de los registros al final de cada secuencia de

instrucciones? ¿Cuál valor es el primero en ser adelantado y cuál valor sobrescribe?

- f) Si usted por error olvidó implementar el hardware para adelantamiento, ¿cuáles son los valores de los registros al final de cada secuencia de instrucciones?
 - g) Agregue *nops* a la secuencia de instrucciones para evitar los problemas vistos en el punto f.
7. Con base en el siguiente código:

```
int n=1024;
for (x=0, x<n, x++) {
    if (x<n/2){
        a[x]=1;
    }
    else{
        a[x]=0;
    }
}
```

- a. Convierta el código mostrado a ensamblador (MIPS).
 - b. Identifique los posibles riesgos en el código de ensamblador para una arquitectura con un pipeline de 5 etapas sin hardware replicado.
 - c. Determine el número de **bloques básicos** que posee el código en ensamblador, así como el contenido en instrucciones de cada bloque.
 - d. Si las instrucciones, en general, toman 5 ciclos de reloj en ejecutarse y considerando que cada bloque básico puede ejecutarse en paralelo mediante un **pipeline balanceado** (no considere dependencias ni riesgos, asuma que la predicción de saltos funciona en un 100% de los casos) ¿cuál sería el WCET (en ciclos de reloj) para el código anterior?
 - e. Considere ahora en la ejecución del código los posibles riesgos encontrados en el punto b) ¿Cuántos **stalls** tendrían que agregarse para evitar dichos riesgos? ¿Cuál sería el nuevo tiempo de ejecución (en ciclos de reloj) del programa?
8. Con base en siguiente código:

```

_init:
    LW   R1, A
    ADD R3, R1, R2
    SLL  R1, R2, 3
    ADD R1, R1, R2
    BEQ  R2, T0, ext
    SUB R1, R2, R3
    SW   R2, A

_ext:
    done

```

- a) Mencione todas las dependencias (datos, nombre, control) existentes en el mismo para una arquitectura con ejecución en orden y una fuera de orden.
- b) Para una arquitectura con un pipeline de 5 etapas y una posible ejecución fuera de orden determine cuáles riesgos se podrían presentar y cómo podrían solucionarse.
- c) Para una arquitectura en orden de 5 etapas de pipeline, muestre la ejecución del código anterior a nivel de etapas considerando riesgos y tomando una política de predicción de saltos de “no tomado”.

CONFIABILIDAD

9. La disponibilidad es la consideración más importante a la hora de diseñar servidores, seguido por la escalabilidad y el flujo de instrucciones.
- a) Se tiene un procesador con un promedio de 100 fallas por año. ¿Cuál es el tiempo medio para falla (MTTF), en días, del sistema?
 - b) Si toma un día para tener el sistema funcionando de nuevo, ¿cuál es la disponibilidad del sistema?
 - c) Suponga que el gobierno, para recortar gastos, va a construir un supercomputador de componentes económicos, en lugar de componentes más caros y confiables. ¿Cuál es el MTTF para el sistema con 1000 de los procesadores mencionados arriba? Asuma que los procesadores fallan simultáneamente.
10. En granjas de servidores usadas por Amazon o eBay, una sola falla no causa que el sistema entero falle. En su lugar, causa la reducción del número de peticiones que pueden ser atendidas simultáneamente.
- d) Si una compañía tiene 10000 computadores, cada uno con un MTTF de 35 días, y experimenta una falla catastrófica solamente si $\frac{1}{3}$ de los computadores fallan, ¿cuál es el MTTF del sistema?
 - e) Si cuesta \$1000 extra, por computador, doblar el MTTF, ¿sería esta una buena decisión de negocio? Justifique su respuesta.