Instituto Tecnológico de Costa Rica Área de Ingeniería en Computadores Profesor: Lic. Luis Alonso Barboza Artavia

Profesor: Lic. Luis Alonso Barboza Artavia CE 4302 Arquitectura de Computadores II

I Semestre 2021

1 Semestre 2021		
Carné:	Nombre:	_ Nota:

INSTRUCCIONES GENERALES.

- Debido al estado de la crisis se recalca la importancia de mantener una comunicación continua y pronta con el profesor para atender cualquier eventualidad lo más rápido posible.
- Si presenta problemas en esta evaluación contacte al profesor por correo electrónico o al teléfono.
- Se realizará una videollamada durante la realización de la evaluación para atender las dudas inmediatamente.
- Esta evaluación es **individual** y tiene una duración máxima de 120 minutos.
- La entrega está habilitada desde el minuto 0 de la evaluación.
- Si tiene problemas subiendo contenido adjunte un screenshot y la justificación al correo electrónico del profesor inmediatamente, recuerde la importancia de la comunicación oportuna.
- Responda de forma clara, ordenada y legible en un pdf. Esta evaluación debe ser escrita a mano, no se permite editores de texto.
- El documento será sometido a control de plagios, se prohíben copias textuales de otros estudiantes o sitios en Internet.
- El documento debe reflejar el entendimiento del concepto, por esta razón tiene que ser explicado en sus propias palabras, sin recurrir a citas bibliográficas.
- Este examen es de 36 puntos.

I. Desarrollo. 36 puntos

A continuación se le presentan una serie de preguntas que debe contestar de forma clara y concisa.

- 1. Explique de manera detallada qué es *Dynamic Voltage Frequency Scaling (DVFS)* y qué consecuencias tiene su uso en HPC. (2 pts)
- 2. Explique de manera detallada cómo un procesador vectorial trabaja con vectores más grandes que la cantidad de elementos en su vector. (3 pts)
- 3. Explique con 4 argumentos porqué se opta la mayoría de las veces por una extensión SIMD en lugar de las otras versiones de SIMD (vectorial y GPU). (3 pts)
- 4. Explique dos ventajas de utilizar computación heterogénea sobre computación homogénea en computación de alto desempeño. (2 pts)
- 5. Explique tres de las características o requerimientos propios de un WSC y cómo afecta cada uno en el diseño e implementación del WSC. (3 pts).
- 6. Considere el siguiente código en VMIPS. Asuma un único lane por unidad funcional.

Examen 2

```
LV V2, Rx

LV V1, Ry

ADDVV.D V3,V2,V1

SUBVV.D V4,V2,V1

LV V5, Rz

MULVS.D V8,V9,F0

ADDVV.D V5,V6,V7

SV V5,Ry

ADDVV.D V3,V2,V1

SUBVV.D V4,V2,V1
```

- 6.1. ¿Cuántos ciclos tomará ejecutar la secuencia? (2 pts)
- 6.2. Determine la cantidad de FLOPS si se trabaja a una frecuencia de 3.3MHz. (2 pts)
- 6.3. Explique de manera detallada cuántos *lanes* pueden maximizar la optimización de la secuencia. (2 pts)
- 7. Suponga que, al ejecutar un programa con una intensidad aritmética de 1 , el computador vectorial tuvo un desempeño de 16 GFLOP/s y el multinúcleo de 128 GFLOP/s. Al ejecutar un programa con una intensidad 3/2, el computador vectorial tuvo un desempeño de 20 GFLOP/s, mientras que el multinúcleo obtuvo 150 GFLOP/s. El desempeño pico de los computadores es 55 GFLOP/s para el vectorial y 175 GFLOP/s para el multinúcleo. A partir de esta información:
 - 7.1. Determine el desempeño de cada computador para un programa con una intensidad aritmética de 1/2. (2 pts)
 - 7.2. Determine el ancho de banda para cada computador. (2 pts)
 - 7.3. Calcule el valor máximo de intensidad aritmética para obtener el desempeño pico en ambos computadores. (2 pts)
 - 7.4. Muestre un dibujo del modelo roofline para cada computador. (2 pts)
 - 7.5. Explique qué indica el computador cuando el valor del rendimiento, según la intensidad aritmética, se encuentra en la pendiente del modelo roofline. (3 pts)
- 8. Considere el siguiente código, que implementa operaciones de punto flotante con precisión simple con números complejos. Suponga que el mismo código se utiliza para comparar desempeño de dos computadores: uno vectorial y uno multiprocesador.

```
for(i=0; i<2048; i++)
{
    c_re[i] = a_re[i] * c_re[i] - a_im[i] * b_im[i];
    c_im[i] = a_re[i] * b_im[i] - c_im[i] * b_re[i];
}</pre>
```

- 8.1. Determine matemáticamente la intensidad aritmética del código. (3 pts)
- 8.2. Agregue líneas de código con el fin de que la intensidad aritmética sea 1/8. (3 pts)