

EVALUACIÓN 3 – 40%

ESTUDIO DE CASO – Cálculo de la Pseudoinversa

Nombre de los Integrantes	
Fecha de Entrega	
Carrera	Ingeniería Civil Informática/Magíster en Ciencias de la Computación
Académico	M. Sc. Sergio Antonio Baltierra Valenzuela
Actividad Curricular	Evaluación 3

Resultados de Aprendizajes Evaluados:	Aplicar técnicas de computación paralela en sistemas many-core.
Indicadores de Evaluación:	Desarrollar algoritmos de programación multi-hilos en GPU para la solución problemas.

PUNTAJE MÁXIMO	35	PUNTAJE DE CORTE	21	PUNTAJE OBTENIDO		CALIFICACIÓN	
----------------	----	------------------	----	------------------	--	--------------	--

INSTRUCCIONES	
<p><u>Se Pide</u></p> <ul style="list-style-type: none"> • Debe desarrollar un algoritmo paralelo usando CUDA C. • Debe hacer un buen balance entre el número de bloques e hilos por bloque en CUDA C. • Debe calcular la métrica speedup. • Debe guardar los resultados de los ensayos y la métrica en un solo archivo. <p><u>Entrega</u></p> <p>La solución se debe entregar en un solo archivo comprimido el cual debe tener la denominación: NOMBRE_APELLIDOPATERNO_APELLIDO_MATERNO.{zip,tar,rar} y subir el archivo al UCM Virtual dentro del plazo definido.</p> <p>Ejemplo de archivo a subir: SERGIO_BALTIERRA_VALENZUELA.{zip,tar,rar}.</p> <p><u>El archivo comprimido debe contener:</u></p> <ul style="list-style-type: none"> • <u>El código paralelo de extensión .cu</u> • <u>El archivo de resultados de nombre “metrica.met”.</u> <p><u>Plazo</u></p> <ul style="list-style-type: none"> • <u>9 de julio hasta las 12:00 horas (UCM Virtual).</u> 	

Integrantes

Se puede trabajar de manera individual o en grupo de máximo dos integrantes los cuales serán conformados por afinidad.

En caso de trabajar en grupo debe denominarse el archivo a subir con los dos integrantes, ejemplo: JUAN_PEREZ_PEREZ_Y_ALAM_BRITO_DELGADO.{zip,tar,rar}

Restricciones

- Se aplicará artículo 67º del reglamento del estudiante, el cual indica que, en caso de sorprender copia parcial o exacta, ya sea entre compañeros o reproducidos de algún medio, lo cual implica un 1,0 para todos los involucrados.
- Si hay algún requerimiento sin contestar, y en la rúbrica no se puede observar, no aplica puntaje.

DESCRIPCIÓN DEL CASO Y DE LAS PREGUNTAS O REQUERIMIENTOS A SER RESUELTOS

Contexto

La empresa “No Compila, S.A.” necesita optimizar algoritmos para sus sistemas usando CUDA C que permite paralizar estos algoritmos. El algoritmo que se debe optimizar es el más importante que se denomina: “Cálculo de la Pseudoinversa”.

Caso

En algebra lineal, una matriz cuadrada A tiene inversa A^{-1} cuando al multiplicarse por ella da como resultado la matriz identidad.

$$A \cdot A^{-1} = A^{-1} \cdot A = I$$

Ahora, que pasa cuando la matriz no es cuadrada, es decir tiene dimensiones $A_{m \times n}$, donde $m \neq n$, m y n pertenecen a los naturales mayor que cero, donde:

1. Existe una matriz $P_{n \times m}$ denominada R , tal que:

$$A_{m \times n} \cdot R = I_m$$

Por lo que R es la pseudoinversa de $A_{m \times n}$ por la derecha.

2. Existe una matriz $P_{n \times m}$ denominada L , tal que:

$$L \cdot A_{m \times n} = I_n$$

Por lo que L es la pseudoinversa de $A_{m \times n}$ por la izquierda.

En ambos casos, pseudoinversa por la derecha y la izquierda, se pueden obtener usando la siguiente ecuación:

$$\text{Pseudoinversa por la } \begin{cases} R = A^t(AA^t)^{-1} \\ L = (A^tA)^{-1}A^t \end{cases}$$

Para verificar si una matriz tiene pseudoinversa, se puede saber usando la siguiente ecuación:

$$\text{Existe pseudoinversa por la } \begin{cases} \text{Derecha, ssi tiene rango}(A_{m \times n}) = m \\ \text{Izquierda, ssi tiene rango}(A_{m \times n}) = n \end{cases}$$

Ejemplo:

Dada la matriz $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$, el $\text{rango}(A) = 2$, por lo que solo tiene pseudoinversa por la izquierda, lo que implica que debemos usar la ecuación $L = (A^tA)^{-1}A^t$. Quedando como resultado final $L = \begin{bmatrix} -\frac{4}{3} & -\frac{1}{3} & \frac{2}{3} \\ \frac{13}{12} & \frac{1}{3} & -\frac{5}{12} \end{bmatrix} = \begin{bmatrix} -1.333333 & -0.333333 & 0.666667 \\ 1.083333 & 0.333333 & -0.416667 \end{bmatrix}$

Entrada

Se debe leer un archivo de texto denominado "entrada.ent", el cual tiene la siguiente estructura:

3	2
1	2
3	4
5	6

Donde:

- La primera línea contiene los valores m y n , que son las dimensiones de la matriz A .
- La segunda línea en adelante contiene la matriz A .

Se pide

1. Debe desarrollar un algoritmo paralelo usando CUDA C el cual calcule la pseudoinversa de una matriz A no cuadrada.
2. Debe medir los tiempos de la solución tanto secuencial como paralela, para poder verificar si su solución es eficiente, debe calcular la métrica *speedup*, cuyas fórmula son:

A.-

$$\text{speedup} = \frac{T_{\text{secuencial}}}{T_{\text{paralelo}}}$$

Donde:

- $T_{secuencial}$ es el tiempo de ejecución secuencial.
- $T_{paralelo}$ es el tiempo de ejecución paralelo.

Observación: use el tiempo secuencial de la evaluación 2.

El tiempo T de ejecución se calcula usando la fórmula:

$$Tiempo_{ejecución} = T_{final} - T_{inicial}$$

Donde:

- El T_{final} es el tiempo antes de comenzar las instrucciones del algoritmo.
 - El $T_{inicial}$ es el después de terminar las instrucciones del algoritmo.
3. Debe calcular la métricas del punto 3, donde los resultados obtenidos los debe almacenar y entregar en un archivo. El nombre del archivo es "metrica.met", y debe tener la siguiente estructura:

Ensayo	Número de Bloques	Número de Hilos por Bloque	Speedup
1	A	2^n	X.XXX...
2	B	2^{n-1}	Y.YYY...
3	C	2^{n-2}	Z.ZZZ...
...

Donde:

- La primera columna es el número del ensayo que debe ser mínimo 10.
- La segunda columna es el número de bloques que se usan en los ensayos y que debe ser un valor entero positivo.
- La tercera columna es el número de hilos por bloque que se usan en los ensayos que debe ser un valor de potencia base 2.
- La cuarta columna es el resultado del *Speedup* por cada ensayo (usar el máximo de decimales posibles).

Salida

El resultado del algoritmo se debe guardar en un archivo de texto de salida denominado "salida.sal", el cual tiene la siguiente estructura:

```
L
-1.333333 -0.333333 0.666667
1.083333 0.333333 -0.416667
```

Donde:

- La primera línea contiene una letra, L para pseudoinversa a la izquierda o R para pseudoinversa a la derecha.
- La segunda línea en adelante contiene la pseudoinversa de la matriz A (usar el máximo de decimales posibles).

Observación: en caso que una matriz no tenga pseudoinversa, debe guardar como resultado un -1 dentro del archivo de salida.

Restricciones

- La matriz A a calcular pseudoinversa debe tener rango completo, es decir, **rango** = **min(m, n)**.
- Máximo de decimales de la matriz A en el archivo de "entrada.ent" son seis.
- Máximo valor de m y n en la matriz A es 100.

Rúbrica

Indicadores	Destacado (5 puntos)	Competente (4 puntos)	Básico (3 puntos)	En desarrollo (1 punto)	Observaciones
Correctitud del Algoritmo Paralelo	El algoritmo paralelo en CUDA C distribuye eficientemente las tareas entre bloques e hilos por bloque (potencias de 2), calcula correctamente la pseudoinversa y mejora el rendimiento sin errores de sincronización.	El algoritmo paralelo en CUDA C distribuye eficazmente las tareas entre bloques e hilos por bloque (potencias de 2), calcula la pseudoinversa y mejora en parte el rendimiento.	El algoritmo paralelo en CUDA C distribuye poco eficazmente las tareas entre bloques e hilos por bloque (potencias de 2), calcula en parte la pseudoinversa y el rendimiento.	El algoritmo paralelo no funciona o no implementa correctamente CUDA C.	
CUDA C	Hace uso correcto de la sintaxis y semántica en CUDA C, realiza un eficiente balance entre el número de bloques e hilos por bloque en cada ensayo.	Hace un buen uso de la sintaxis y semántica en CUDA C, realiza un eficaz balance entre el número de bloques e hilos por bloque en cada ensayo.	Hace uso de la sintaxis o semántica con leves errores o problemas en CUDA C, realiza un poco eficaz balance entre el número de bloques e hilos por bloque en cada ensayo.	Hace uso de la sintaxis o semántica con muchos errores o problemas en CUDA C, realiza un mal balance entre el número de bloques e hilos por bloque en cada ensayo.	
Archivos de entrada y salida	El programa lee y escribe los archivos con el formato especificado perfectamente en el enunciado y con el nombre correcto	El programa lee y escribe archivos correctamente pero con un error en el formato y con el nombre correcto "entrada.ent" y "salida.sal".	El programa lee y escribe archivos pero tiene varios dos errores en el formato. El nombre no es el correcto "entrada.ent" y "salida.sal".	El programa no lee y escribe archivos en el formato especificado. El nombre no es el correcto "entrada.ent" y "salida.sal".	

	"entrada.ent" y "salida.sal".				
Buenas prácticas de programación	El código está bien estructurado, con comentarios claros, nombres de variables significativos, y sigue las buenas prácticas de programación.	El código está mayormente bien estructurado, con algunos comentarios y nombres de variables significativos, y sigue las buenas prácticas de programación.	El código es funcional pero tiene varios problemas de estructura y falta de comentarios, el código sigue parcialmente las buenas prácticas de programación.	El código es difícil de seguir, con poca o ninguna estructura y comentarios, el código no sigue por completo las buenas prácticas de programación.	
Cumplimiento del plazo de entrega	Entregado dentro del plazo definido en el UCM Virtual. Además, entrega en el formato solicitado, incluyendo todos los elementos solicitados y los elementos propios de nombre y tipo de archivos.	Entregado con 5 a 30 minutos de atraso en el UCM Virtual. Además, entrega en el formato solicitado, incluyendo la mayoría de los elementos solicitados y la mayoría de los elementos propios de nombre y tipo de archivos.	Entregado con 31 a 60 minutos de atraso en el UCM Virtual. Además, entrega en el formato solicitado, incluyendo algunos de los elementos solicitados y los algunos de los elementos propios de nombre y tipo de archivos.	Entregado con 61 o más de la fecha límite en UCM Virtual. Además, entrega en el formato solicitado, incluyendo pocos de los elementos solicitados y pocos elementos propios de nombre y tipo de archivos.	
Optimización	El programa está altamente optimizado, usando paralelismo efectivamente con CUDA C para reducir significativamente el tiempo de ejecución.	El programa usa paralelismo con CUDA C pero tiene algunas áreas que podrían estar mejor optimizadas.	El programa usa paralelismo de manera básica pero no está bien optimizado.	El programa tiene intentos de paralelismo pero no mejora significativamente el rendimiento.	

Cálculo de Métrica Speedup	Calcula correctamente speedup para al menos 10 ensayos con bloques e hilos por bloque en potencias de 2, usando las fórmulas proporcionadas, y los resultados son consistentes con el rendimiento del algoritmo paralelo. Todo en el archivo debe ser entregado con el formato y nombre según el enunciado "metrica.met".	Calcula speedup correctamente, pero con errores menores en los cálculos o en menos de 10 ensayos, o los resultados son ligeramente inconsistentes. Todo en el archivo debe ser entregado con el formato y nombre según el enunciado "metrica.met".	Calcula la métrica, pero con errores significativos en las fórmulas o en la cantidad de ensayos, afectando la validez de los resultados. El archivo no tiene el formato y/o nombre según el enunciado "metrica.met".	Calcula la métrica incorrectamente o no realiza suficientes ensayos, con resultados poco confiables. El archivo no tiene el formato y nombre según el enunciado "metrica.met".	
----------------------------	---	--	--	--	--

Escala de notas

Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota
0.0	1.0	10.0	2.4	20.0	3.9	30.0	5.9
1.0	1.1	11.0	2.6	21.0	4.0	31.0	6.1
2.0	1.3	12.0	2.7	22.0	4.2	32.0	6.4
3.0	1.4	13.0	2.9	23.0	4.4	33.0	6.6
4.0	1.6	14.0	3.0	24.0	4.6	34.0	6.8
5.0	1.7	15.0	3.1	25.0	4.9	35.0	7.0
6.0	1.9	16.0	3.3	26.0	5.1		
7.0	2.0	17.0	3.4	27.0	5.3		
8.0	2.1	18.0	3.6	28.0	5.5		
9.0	2.3	19.0	3.7	29.0	5.7		