**Hoja de trabajo #3 - Computación Paralela y Distribuida**

**Repositorio de GitHub:** [[[erickguerra22/HDT3\_Paralela](https://github.com/erickguerra22/Corto2_Paralela.git)](https://github.com/erickguerra22/HDT3_Paralela.git)](https://github.com/erickguerra22/HDT3_Paralela.git)

**Ejercicio 1**

El programa hello.cu ilustra la forma básica del modelo de ejecución para CUDA. Realice las siguientes acciones para comprender el efecto de la configuración del kernel y su relación con el Compute Capability de una tarjeta.

1. Compile el programa (ignore la advertencia sobre código deprecado en caso le salga):
2. Ejecute el programa. Observe cuántas veces se imprime el mensaje y su conexión con la configuración de la llamada al kernel − hello<<<g,b>>>():

*Imagen 1.1: Ejecución de ‘hello.cu’*Texto

Descripción generada automáticamente

**R:** El mensaje se ha impreso un total de 10 veces, lo cual va de acuerdo con la instrucción hello<<<1,10>>>(), en donde se indica que se ejecute la función en 1 bloque, dividido en 10 hilos.

1. Modifique el programa para correr 2 bloques de 1024 hilos. Modificarlo también para que imprima su nombre y carnet. Busque en el despliegue de consola el mensaje del último hilo de la serie (1023).

*Imagen 1.2: Penúltimo hilo de primer bloque*Imagen que contiene marcador, azul, cerca, reloj

Descripción generada automáticamente

*Imagen 1.3: Penúltimo hilo de segundo bloque  
Imagen de la pantalla de una computadora

Descripción generada automáticamente con confianza baja*

1. Busque en el sitio de Nvidia el Compute Capability de la tarjeta que poseen las máquinas del Laboratorio (o de la computadora que está utilizando). Escriba acá el valor de CC y busque la tabla resumen con las características técnicas del CC:

**R:** Para este laboratorio se está utilizando Google Colaboratory, cuya máquina utiliza la tarjeta gráfica NVIDIA Tesla T4, cuya Compute Capability es de 7.5. Las especificaciones técnicas básicas para este tipo de Compute Capability se muestran a continuación:

Captura de pantalla de computadora

Descripción generada automáticamente

Para ver la tabla completa e información más detallada, dirigirse a: [CUDA C++ Programming Guide/Compute Capabilities/Features and Technical Specifications](https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#features-and-technical-specifications)

1. Modifique el programa para correr 1 bloque de 2048 hilos.

*Imagen 1.4: Ejecución con 1 bloque y 2048 hilos  
Texto

Descripción generada automáticamente*

**R:** Como se aprecia en la imagen, el código no fue capaz de imprimir ningún resultado, ya que tal como se muestra en la tabla anterior con las especificaciones técnicas para la Compute Compatibility de 7.5, el máximo de hilos soportados por bloque es de 1024. Por lo tanto, la tarjeta gráfica no fue capaz de procesar la solicitud de 1 bloque con 2048 hilos.

1. Busque en la tabla de CC los siguientes datos de la GPU que está utilizando:
   1. Warp size: **32**
   2. Maximum number of threads per block: **1024**
   3. Maximum dimensionality of a grid of thread blocks: **3**
   4. Maximum size per grid dimension:
      1. X: **2^31-1**
      2. Y, Z: **65535**
   5. Maximum dimensionality of a thread block: **3**
   6. Maximum size per block dimension:
      1. X,Y: **1024**
      2. Z: **64**

**Ejercicio 2**

El programa hello2.cu ilustra la forma para calcular un identificador global al momento de usar hilos que pertenecen a bloques diferentes. Realice las siguientes acciones para comprender el efecto de la configuración del kernel y su relación con la forma de calcular el ID único de los hilos.

1. Descargue, compile y ejecute hello2.cu. Observe la relación de la configuración de la llamada al kernel con la geometría de los hilos y el resultado. Escriba la respuesta a los dos enunciados:

*Imagen 2.1: Ejecución de ‘hello2.cu’*  
Texto

Descripción generada automáticamente

* 1. Máximo ID de los hilos: **239**
  2. Ejecución de los hilos en orden: **Si bien la impresión de los hilos no va en orden secuencial desde 0 hasta 239, sí siguen secuencias de decenas (0-9), esto se debe a que el cálculo del ID del hilo depende estrictamente del bloque en el que se están ejecutando. Al asignarse 10 hilos para cada bloque, los IDs de un mismo bloque respetarán la secuencia de 0 a 9.**

1. Observe que la fórmula genérica para cálculo del ID global está en los comentarios. Modifique el programa para que imprima también su nombre y carné. Luego, realice la siguiente modificación al programa (al inicio del main) y use la fórmula genérica para derivar el nuevo cálculo de ID:

dim3 g (4,2);  
dim3 b (32,16);  
hello <<>>();

*Imagen 2.2: Fórmula para calcular el ID global  
Texto

Descripción generada automáticamente*

*Imagen 2.3: Salida de pantalla con fórmula descomentada  
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente*

*Imagen 2.4: Nueva configuración  
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media*

*Imagen 2.5: Hilo máximo con nueva configuración  
Texto

Descripción generada automáticamente*

1. Revise nuevamente la información del Compute Capability respecto a las dimensiones máximas de hilos-bloque en x, y, & z para una grilla. Cree una configuración para lanzar exitosamente el kernel para procesar 100,000 datos. (Sugerencia: busque una configuración que lance como mínimo 100,000 hilos. Modifique el kernel para que imprima el mensaje únicamente si es el ID global máximo)

*Imagen 3.1: Nueva configuración  
Captura de pantalla de computadora

Descripción generada automáticamente*