TEC | Tecnológico de Costa Rica

Instituto Tecnológico de Costa Rica
Área Académica de Ingeniería en Computadores
CE 4302 — Arquitectura de Computadores II

Taller 4:
CUDA

Fecha de asignación:	28 de abril 2023	Fecha de entrega:	05 de mayo 2023
Grupo:	4 personas (proyecto 2)	Profesor:	Luis Barboza Artavia

1. Investigación

Para comprender mejor CUDA, realice una pequeña búsqueda para responder las siguientes preguntas:

1. ¿Qué es CUDA?
2. ¿Qué es un kernel en CUDA y cómo se define?
3. ¿Cómo se maneja el trabajo a procesar en CUDA? ¿Cómo se asginan los hilos y bloques?
4. Investigue sobre la plataforma Jetson Nano ¿cómo está compuesta la arquitectura de la plataforma a nivel de hardware?
5. ¿Cómo se compila un código CUDA?

2. Requisitos

El día del taller deben traer lo siguiente:

1. microSD (mínimo 16GB) con la imagen de la Jetson Nano.
2. Cable micro-USB.

3. Previo al taller

Antes de iniciar las actividades de forma presencial en el laboratorio debe realizar las siguientes tareas.

1. Descargar la imagen (enlace) de la Jetson Nano.
2. Escribir la imagen en la microSD. Se recomienda Etcher.

4. Configuración Jetson Nano

Una vez en el laboratorio, realizar los siguientes pasos de configuración.

1. Colocar la microSD en la Jetson Nano. En la Figura 1 se muestra dónde está la ranura para la microSD.

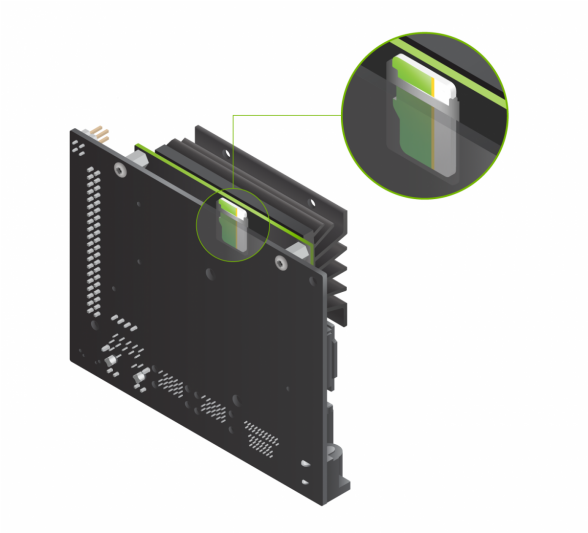

Figura 1: Colocación de microSD.

2. Conectar la Jetson Nano. Si utiliza el monitor, se requiere micro-USB o un conector de 5V. Si utiliza PuTTy, se requiere conector de 5V.
3. Verificar que a la par del conector micro-USB se encienda un LED verde.
4. Configurar el equipo de acuerdo con las instrucciones mostradas en pantalla.
5. Verificar la versión de CUDA instalada. Para esto, ejecutar el siguiente comando: `nvcc --version`.
6. En caso de que no encuentre CUDA, verificar el archivo `~/.bashrc` contenga las siguientes líneas. Si no se encuentran, agregarlas y reiniciar la terminal o sesión.

```
export PATH=/usr/local/cuda/bin${PATH:+:${PATH}}
export LD_LIBRARY_PATH=/usr/local/cuda/lib64${LD_LIBRARY_PATH (continua)
:+:${LD_LIBRARY_PATH}}
```

7. Realizar de nuevo el paso 5.

5. Análisis

Como primera prueba de CUDA, se va a trabajar con el archivo *vecadd.cu*, que utiliza un *kernel* en GPU para una operación de suma. Antes de continuar, debe asegurarse de entender correctamente qué es un kernel y cómo funcionan en el ámbito de CUDA.
Con base en el código de ejemplo:

1. Analice el código vecadd.cu.
2. Analice el código fuente del kernel vecadd.cu. A partir del análisis del código, determine ¿Qué operación se realiza con los vectores de entrada? ¿Cómo se identifica cada elemento a ser procesado en paralelo y de qué forma se realiza el procesamiento paralelo?
3. Realice la compilación del código vecadd.cu.
4. Realice la ejecución de la aplicación vecadd. ¿Qué hace finalmente la aplicación?
5. Cambie la cantidad de hilos por bloque y el tamaño del vector. Compare el desempeño antes al menos 5 casos diferentes.

6. Ejercicios prácticos

1. Realice un programa que calcule el resultado de la multiplicación de dos matrices de 4x4, utilizando paralelismo con CUDA.
2. Proponga una aplicación que involucre procesamiento vectorial. Implemente dicha aplicación tanto serial (sin paralelismo) como con CUDA. Mida tiempos de ejecución para diferentes tamaños y/o iteraciones.

7. Entregables

Se debe de subir en la sección de Evaluaciones los siguientes archivos en una carpeta comprimida: código fuente con la solución de los problemas, README con las instrucciones necesarias para compilar los archivos y un PDF con las respuestas de la investigación y análisis.

Si tienen dudas puede escribir al profesor al correo electrónico. **Los documentos serán sometidos a control de plagios**. La entrega se debe realizar por medio del TEC-Digital en la pestaña de evaluación. No se aceptan entregas extemporáneas después de la fecha de entrega a las 23:59 como máximo.