UNIVERSIDAD DE ALICANTE

CUDA WORKSHOP 2019

PRÁCTICA 1: Suma de Vectores

Sergio Orts Escolano (sorts@ua.es)

Alberto García García (agarcia@dtic.ua.es)

José García Rodríguez (jgarcia@dtic.ua.es)

PRÁCTICA 1 SUMA DE VECTORES

En esta práctica llevaremos a cabo una simple suma de vectores, la cual nos ayudará a entender la estructura de un kernel CUDA. Tendrás que completar porciones de código utilizando el lenguaje CUDA para computar este cálculo ampliamente utilizado. En este laboratorio aprenderemos:

- Aprender a crear un proyecto CUDA con Visual Studio.
- Conocer el flujo típico de un programa CUDA.
- Implementar e invocar un kernel sencillo.
- Utilizar los índices de hilos y de bloques en el kernel para distribuir trabajo entre los hilos.

Conocer limitaciones de la arquitectura mediante el uso de deviceQuery.

Esta práctica se realizará de forma guiada en el laboratorio de forma que todos construiremos poco a poco los tres primeros pasos. Las soluciones a los pasos cuatro, cinco y seis no serán guiadas. El código base hasta el paso tres inclusive se encuentra en el repositorio “practical_assignments/assignment00-cuVectorAdd”.

SUMA DE VECTORES

Paso 1: Editar el punto de entrada del programa main(...) en el fichero main.cu para completar el flujo de ejecución CUDA:

- Declarar los arrays para entrada y salida tanto en CPU como en GPU utilizando cudaMalloc(...).
- Transferir datos de los arrays en CPU a los arrays en GPU utilizando cudaMemcpy(...).

- Definir los parámetros de invocación del kernel (dimensión de bloque y dimensión de grid) e invocar al kernel previamente implementado utilizando esos parámetros.
- Transferir los resultados del array en GPU al array en CPU empleando cudaMemcpy(...).
- Verificar resultados y liberar memoria CPU y GPU con cudaFree(...)

Paso 2: Editar la función suma_vectores(...) en el fichero main.cu para completar el kernel e implementar la funcionalidad de la suma de vectores en la GPU. En esta primera versión se espera que cada hilo de ejecución en la GPU sume un único elemento de los vectores y deposite el resultado en el vector de salida en la posición correspondiente. Así pues, el ejercicio se puede reducir a dos pasos:

- Definir los índices del elemento a ser sumado por cada hilo, empleando las variables especiales de CUDA que permiten obtener los índices de hilo (threadIdx) y de bloque (blockIdx).
- Sumar las posiciones de los vectores de entrada correspondientes al hilo en cuestión, teniendo en cuenta que cada hilo va a computar la suma de un elemento, y depositar el resultado en la posición adecuada del vector de salida.

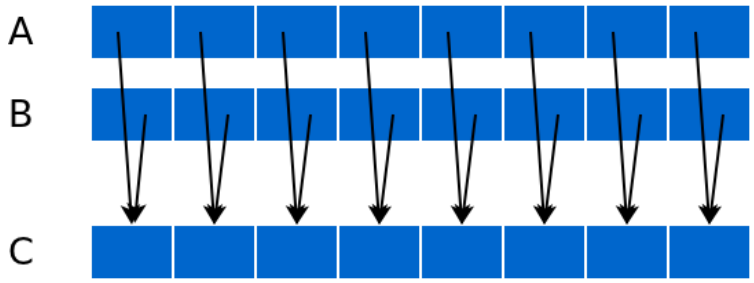

FIGURA 1. Ejemplo suma de vectores.

Paso 3: Ejecutar el código y comprobar que el resultado es correcto.

Paso 4: Modifica el número de elementos a sumar (por ejemplo, 25601), compila y ejecuta de nuevo. ¿Qué está ocurriendo?

Paso 5: Modifica el programa anterior para que se puedan sumar vectores de un tamaño arbitrario distinto al dado.

> **Pista:** El número de elementos era divisible por el número de hilos por bloque
>
> **Pista 2:** No aumentes el número de elementos, simplemente prueba con un número que no sea divisible por el número de elementos por bloque.

Paso 6: Modifica el programa anterior para que se puedan sumar vectores de un tamaño muy grande, prueba con 20.000.000 de elementos.

> **Pista:** Recuerda que las variables número de hilos por bloque y número de bloques tienen un máximo establecido por CUDA.
>
> **Pista 2:** ¿Puede un hilo procesar más de un elemento?

A continuación se listan una serie de documentos y enlaces sobre programación CUDA.

Uno de los primeros cursos sobre programación en CUDA impartido por la universidad de Illinois:

The first course talking about CUDA programming in the world:

http://courses.ce.illinois.edu/ece498/al/

El manual oficial de programación CUDA de NVIDIA, el cual describe el modelo de programación, la sintaxis, una serie de consejos básicos para obtener más rendimiento, especificaciones técnicas, etcétera.

NVIDIA CUDA Programming Guide Version 9.1

http://developer.nvidia.com/nvidia-gpu-computing-documentation

Página oficial de NVIDIA:

CUDA Getting Started Guide (Windows)

Getting Started With CUDA SDK Samples

CUDA Toolkit 9.1 Release Notes & Erratas

CUDA C Programming Guide

Documento de buenas prácticas

http://docs.nvidia.com/cuda/cuda-c-best-practices-guide/

Cursos de programación en CUDA online

https://eu.udacity.com/course/intro-to-parallel-programming--cs344

https://www.coursera.org/course/hetero