CDAP – Práctica 7
Campo de asteroides en CUDA

En esta práctica se simulará un campo de asteroides que interactúan dos a dos mediante atracción gravitatoria. Para generar datos con los que trabajar, se facilita un programa `GenerateAsteroids.c` que creará un fichero `data.txt` con datos de asteroides generados aleatoriamente. Con el enunciado se adjunta un fichero `data.txt` con datos de 1024 asteroides situados en un área bidimiensional de 20 km de lado con masas entre 500 y 10 millones de toneladas.

Se facilita también un programa en C (`SimulateAsteroidsSerial.c`) que realiza la simulación utilizando los recursos de la CPU. Las pruebas realizadas dan unos tiempos de ejecución por encima de los quince minutos para 20000 iteraciones. Si cada iteración equivale a un segundo de simulación, estaríamos haciendo una la evolución del sistema durante poco más de cinco horas y media.

Apartado A (obligatorio)

Se propone la implementación de la simulación del campo de asteroides haciendo uso de los recursos computacionales de la GPU. Para ello, en esta primera versión, se usará un kernel compuesto de un solo bloque unidimensional de N hilos (N es el número de asteroides).

Cada hilo del kernel se encargará de un asteroide. Se recomienda implementar el bucle principal de la simulación fuera del kernel, esto es, cada llamada al kernel se corresponderá con una iteración.

Las pruebas preliminares del programa en el sistema del CITI dan unos tiempos de ejecución aproximados de en torno a los tres minutos (1024 objetos y 20000 iteraciones).

El programa deberá ser subido al ejercicio de faitic (junto con cualquier otro fichero que se desee añadir) en formato .zip o .tar.gz. El nombre del fichero será **PR7aGR#.zip** ó **PR7aGR#.tar.gz** (# es el número del grupo). **Debe ser subido por todos los miembros del grupo de prácticas**.

Apartado B (obligatorio)

Para aumentar el grado de paralelismo, se implementará un kernel en el cual cada hilo se encargará de calcular la interacción entre dos asteroides. Cada hilo calculará las variaciones de la aceleración en los ejes x e y y las acumulará en las velocidades. Comentarios al respecto de esta implementación:

1. No es posible realizar el cálculo usando un kernel con un solo bloque, pues supera el límite máximo de hilos por bloque de la GPU. En el caso de 1024 asteroides, se recomienda usar un grid bidimensional de 32x32 bloques cada uno de los cuales estará compuesto de 32x32 hilos.

2. Cada hilo calculará las aceleraciones ax y ay, y las acumulará en las velocidades vx y vy de los asteroides. Para hacer esto, es necesario usar una operación atómica (atomicAdd) que no está implementada para variables de tipo double. Se facilita el código de una función que se puede invocar sin problema desde el kernel.

La implementación así realizada da unos tiempos de ejecución en el equipo del CITI de en torno a 50 segundos.

El programa deberá ser subido al ejercicio de faitic (junto con cualquier otro fichero que se desee añadir) en formato .zip o .tar.gz. El nombre del fichero será **PR7bGR#.zip** ó **PR7bGR#.tar.gz** (# es el número del grupo). **Debe ser subido por todos los miembros del grupo de prácticas**.

Apartado C (optativo)

En la implementación del apartado anterior, cada hilo se encarga de calcular una interacción, tanto en el eje x como en el eje y. Se puede optimizar aún más el programa aumentando el grado de paralelismo.

Se propone la implementación de un kernel en el cual cada bloque sea tridimensional, de dimensión igual a 2 en el eje z. Los hilos con z=0 se encargarán de las proyecciones de las interacciones en el eje x, mientras que los hilos con z=1 lo harán de las proyecciones en el eje y. Si se usan bloques de 32x32 hilos, se habrá alcanzado el máximo de hilos por bloque, por lo que será necesario replantearse las dimensiones de cada bloque.

Una alternativa es, en vez de usar bloques tridimensionales, hacer uso de un grid tridimensional. Por ejemplo, para el caso de 1024 objetos, podríamos crear un grid de 32x32x2 bloques, en el cual los bloques con z=0 se encargasen de una dimensión y los bloques con z=1 de la otra.

Aspectos a tener en cuenta

- El programa deberá ser enviado por correo electrónico (junto con cualquier otro fichero que se desee añadir) en formato .zip ó .tar.gz. El nombre del fichero será **PR7c.zip** o **PR7c.tar.gz**.
- Este apartado de la práctica **será evaluado de manera individual**.