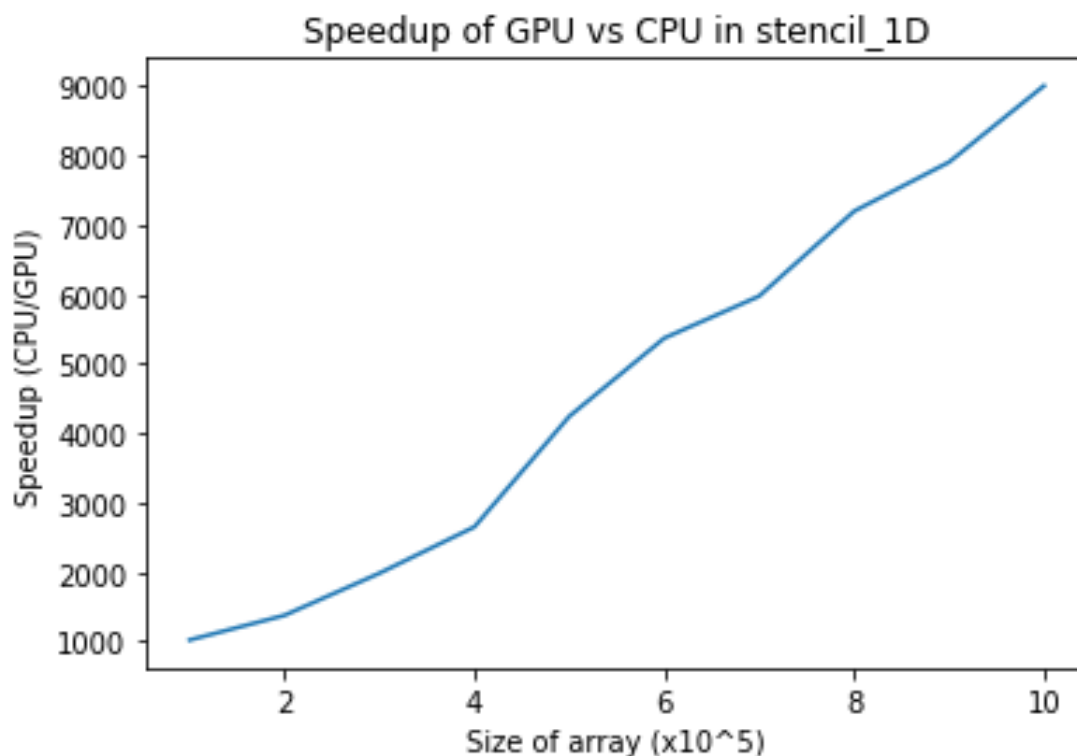


ASIGNATURA Computación de altas prestaciones

Task 2 GPU Programming

You must write a report answering the questions proposed in each exercise, plus the requested files. Submit a zip file through Moodle. Check submission date in Moodle (deadline is until 11:59 pm of that date).

- Exercise 1:
 - o Compare the execution time for different values of N (array size): from 100.000 to 1.000.000 in steps of 100.000. Plot the result in a graph. Explain the results.



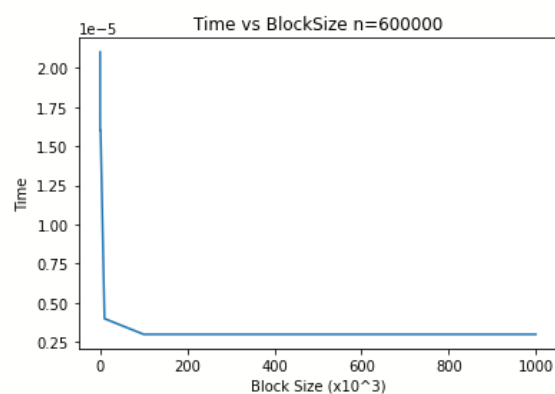
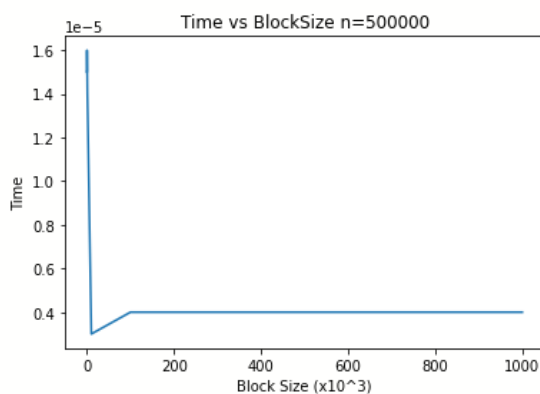
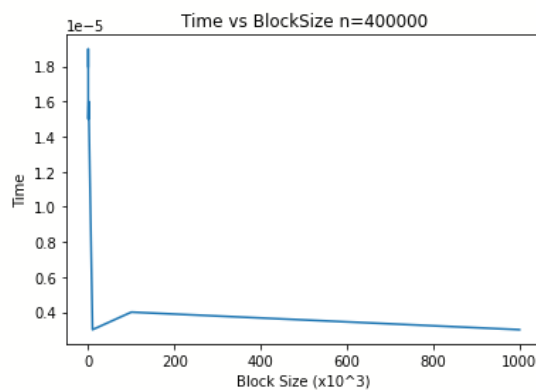
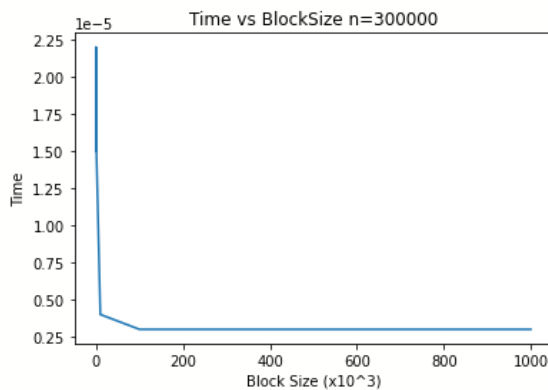
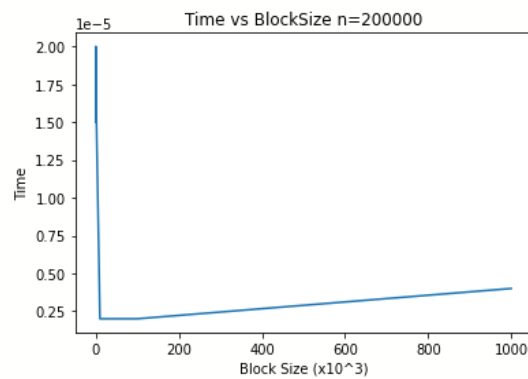
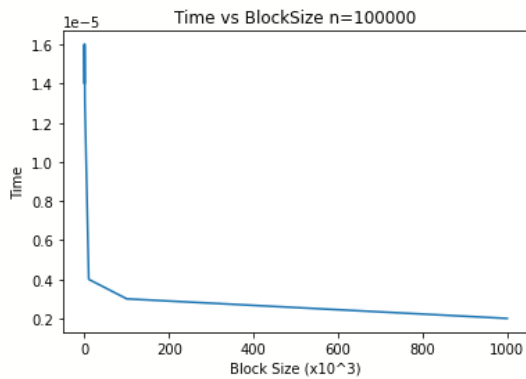
Como se puede ver en los resultados de la ejecución, la aceleración conseguida usando la GPU frente a la CPU es del orden de los miles y continúa en ascenso a medida que el trabajo paralelizable aumenta.

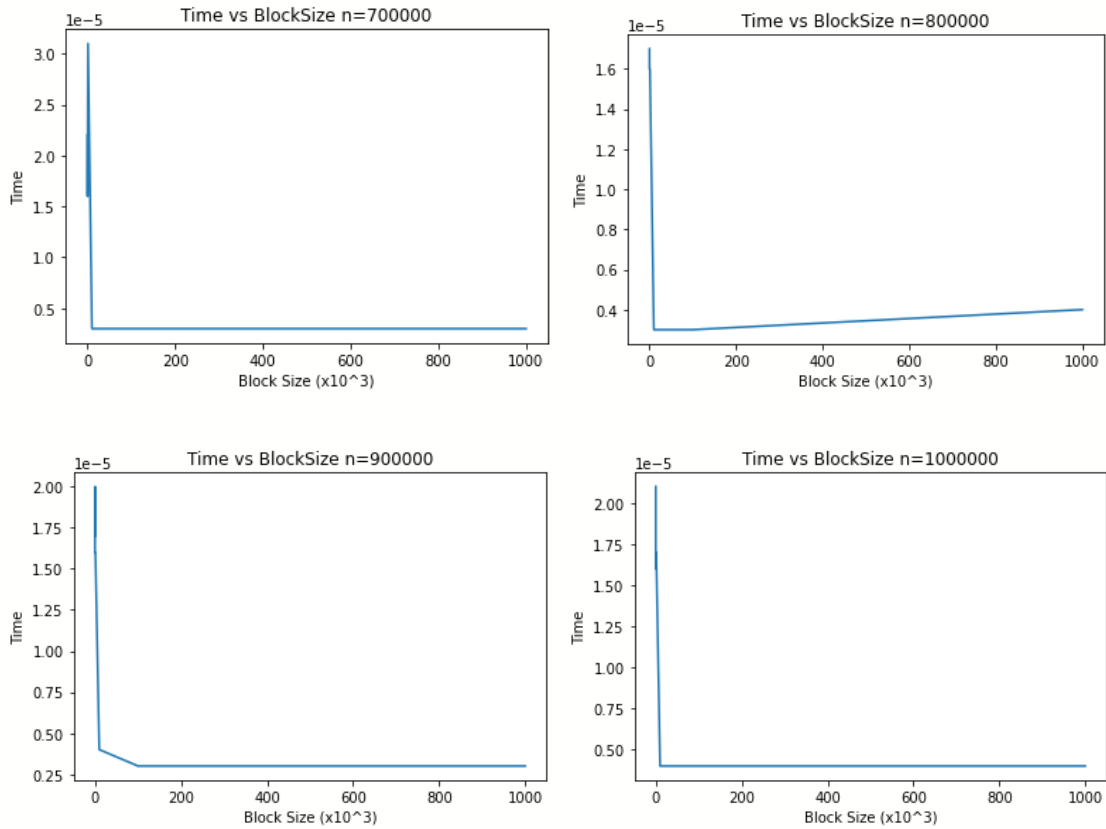
Esto sucede porque la CPU, a pesar de tener una gran potencia de procesamiento, no está pensada para realizar tareas de paralelización masiva como sí lo está la GPU. La GPU consta de muchos más cores que la CPU que, aunque tienen menor capacidad de procesamiento, están especializados en estas tareas de paralelización; obteniendo así unos rendimientos muchísimo mejores que la CPU, como se ha demostrado.

Computación de altas prestaciones HPC

- o What BLOCK SIZE (number of threads per block) have you used? Do you think it is the most optimal? Explain.

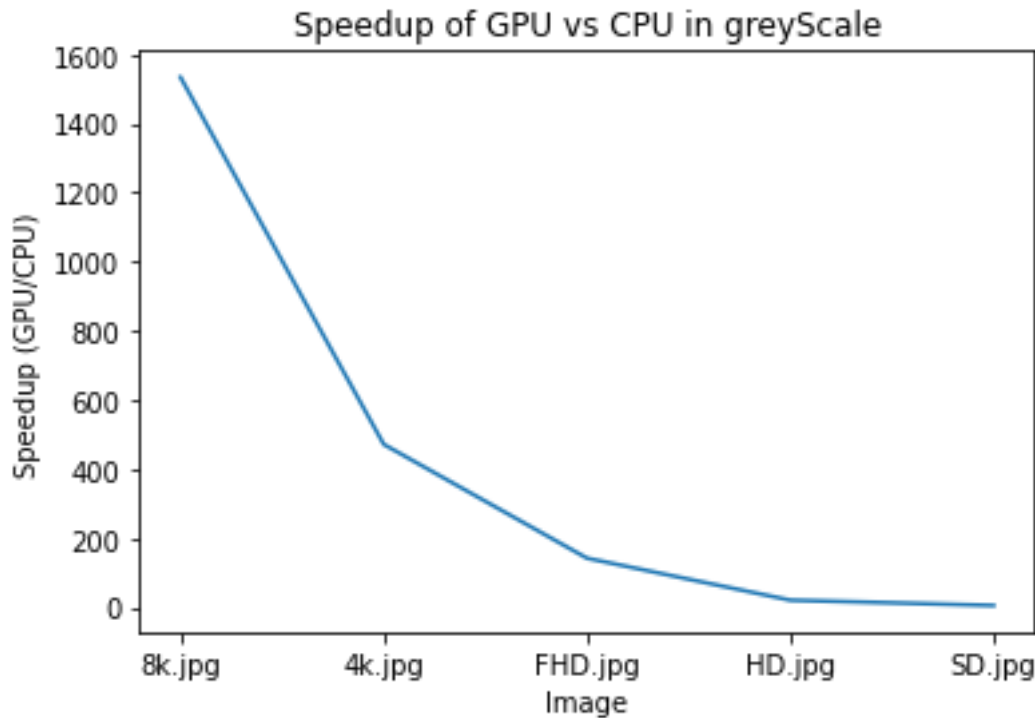
Para el ejercicio se ha fijado el BLOCK SIZE en el tamaño mínimo del array y, para comprobar si era óptimo, se hizo un pequeño programa que comprueba los tiempos logrados con distintos tamaños de bloque comenzando en 1 y multiplicando por 10 en cada iteración para un mismo tamaño de problema. Se repitió el experimento para los mismos tamaños del apartado anterior, de esta forma hemos obtenido los siguientes resultados:





En vista a estos resultados, se puede observar que, alcanzado un valor mínimo de BLOCK SIZE, los resultados temporales parecen ser en mayor o menor medida estables, es por ello que concluimos como óptimo el uso del menor tamaño del array para tamaño de bloque en el ejercicio anterior.

- Exercise 2:
 - o Fill in a table with time and speedup results compared to your manually vectorized CPU code for images of different resolutions (SD, HD, FHD, UHD-4k, UHD-8k). You must include a column with the fps at which the program would process. Discuss the results.



Como se puede ver, cuanto mayor es la calidad de la imagen y, por tanto, su tamaño; mayor es el SpeedUp logrado al paralelizar con Cuda.

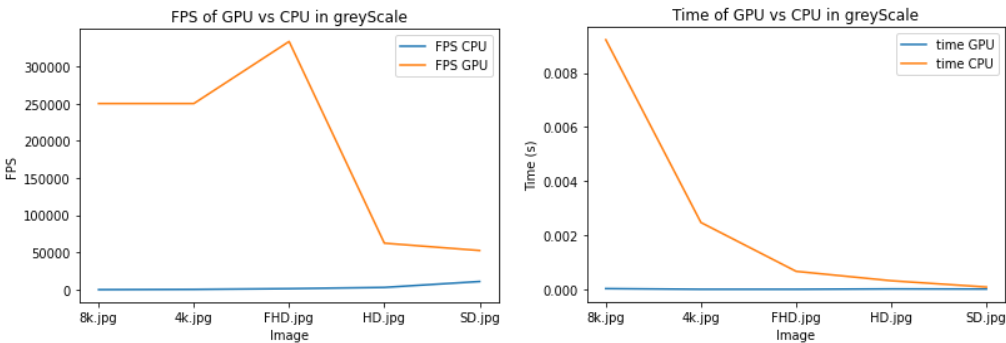


Image	Time CPU	Time GPU	FPS CPU	FPS GPU	SpeedUp
8k.jpg	0.010139	6e-06	98.629	166666.67	1689.83
4k.jpg	0.002481	4e-06	403.0633	250000.0	620.250
FHD.jpg	0.000661	5e-06	1512.869	199999.9	132.2
HD.jpg	0.000319	1.6e-05	3134.796	62500.0	19.9375
SD.jpg	0.000195	1.6e-05	5128.2051	62500.0	12.1875

- o Explain how you implement the algorithm to be optimal for GPU

Dividimos la imagen en altura y anchura, por cada pixel de altura formamos un bloque que lanzará tantos threads como pixels tenga la imagen de ancho, de esta manera, cada pixel tiene asignado un thread encargado de resolver su transformación a gris, como no hay dependencias lo podemos realizar sin preocuparnos por condiciones de carrera, dependencias y otros problemas derivados de la paralelización.