

PCM – Assignment 2

Environment:

All tests were performed in the same environment, a computer equipped with an 8 core (all locked at 4.2 GHz), 16 threads AMD Ryzen 7 3700X and an Nvidia GeForce RTX 2070 Super with a base core clock of 1605MHz. I used the latest version of Visual Studio 2019 and the background processes were kept to a minimum.

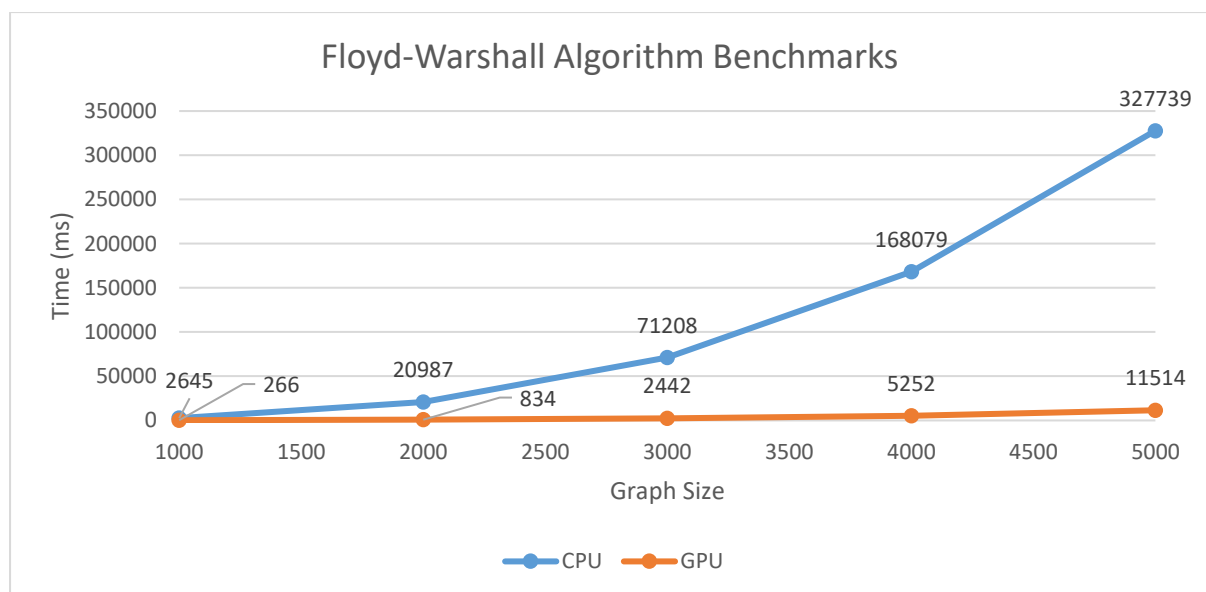
Floyd-Warshall Algorithm:

Taking advantage of the matrix like structure of the GPU, I removed the two for cycles and used two variables (i.e. col and line) using them as coordinates to a variable of the matrix. The program will try to execute the algorithm if the coordinates match a point present in the matrix. Each thread measures the shortest path to the point shown by the coordinates. Using the sum of the points (col, k) + (k, line), is then compared with the value present in the point (col, line), replacing it if the sum is lower than the value present.

To reduce the overhead, the value of col and line is increased by the multiplication of blockDim and gridDim. The thread will execute the algorithm again in case the coordinates match a point present in the matrix.

Regarding memory, it was reserved in the GPU the memory necessary to allocate the graph. The graph located in the CPU was then copied to that memory block, followed by the execution of the algorithm and then the graph located in the GPU was copied back to the CPU. The memory was then freed from the GPU.

The number of threads per block used was the maximum allowed, since the program only used the thread if it was required. Regarding the number of blocks used, a 2D area was created so that the coordinates of each thread would match one point in the graph



On average the program managed a speed up of 24.95 times.