Exercise 6: fast transpose

Student: **Laura Balasso**

Academic year: 2018/19

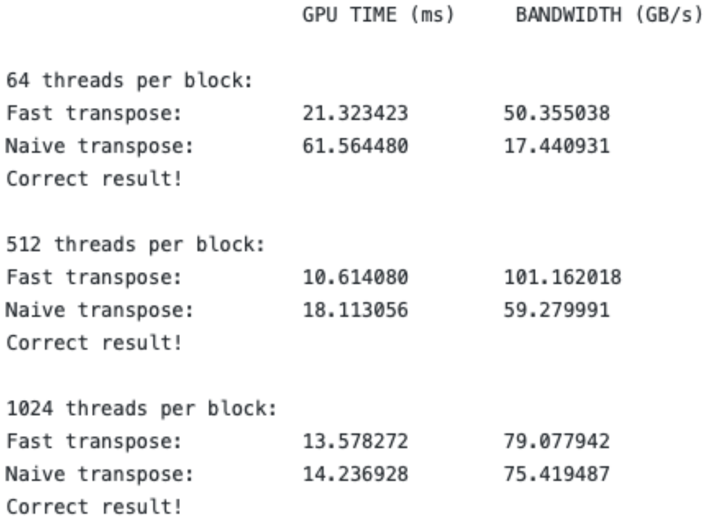
The aim of this exercise is to implement a naïve and an optimized matrix transposition in Cuda, measuring the gpu time for both functions and the associated bandwidth. The exercise asks to run the kernels using a matrix of size 8192x8192 with different numbers of threads per block, namely 64, 512 and 1024.

In both the naïve and the optimized versions, each block of threads transposes a submatrix of size 32x32, thus some threads handle more than one element. Since the exercise asks to use blocks of threads of different sizes, I decided to keep the blocks' columns fix to 32 (in order to match the number of columns of the submatrix) and vary the number of rows, that are computed as below.

$$Block\ Rows = \ Threads\ per\ Block/Block\ Columns$$

The optimized kernel allocates a tile of size 32x32 in the shared memory. Each block uses this buffer to transpose its 32x32 submatrix. With a first for loop, the function reads row-wise from the input matrix and copies the element in the shared memory buffer, then the threads are synchronized to ensure that all the threads copied the values in the tile. Now the indexes can be swapped and with a second loop the function reads column-wise from the shared memory buffer and writes row-wise in the output matrix.

In the image below we can see the execution time in milliseconds of both kernels with different block size, the bandwidth in gigabytes per second and the output of the function which tests the correctness of the matrix transposition by comparing the naïve and the optimized results.



```
                         GPU TIME (ms)     BANDWIDTH (GB/s)

64 threads per block:
Fast transpose:          21.323423        50.355038
Naive transpose:         61.564480        17.440931
Correct result!

512 threads per block:
Fast transpose:          10.614080        101.162018
Naive transpose:         18.113056        59.279991
Correct result!

1024 threads per block:
Fast transpose:          13.578272        79.077942
Naive transpose:         14.236928        75.419487
Correct result!
```

The best results are produced by the fast transpose with 512 threads per block reaching a bandwidth of 101 GB/s.