

## CUDA Exercises

1. Complete the file "hello\_world.cu", so that a single thread was created, with no task, whereas the `main` prints the message.

2. Complete the file "pi\_par\_loop.cu", so that each thread accumulates all its calculations on a local variable and the result is obtained by accumulating all these values.

First compute the reduction in each warp by means of a single loop, and use atomic operations to accumulate the local result on final pi.

Execute the code in patan changing the number of threads, from 256 to 1024 using powers of 2, also changing the number of blocks, from 1 to the maximum number of blocks of the problems, and the number of rectangles {1e7, 5e7, 1e8, 5e8}.

Show the speed-ups obtained and your conclusions.

3. Complete the file "pi\_par\_unroll.cu", so that each thread accumulates all its calculations on a local variable and the result is obtained by accumulating all these values.

First compute the reduction in each warp unrolling the loop, and use atomic operations to accumulate the local result on final pi.

Execute the code in patan changing the number of threads, from 256 to 1024 using powers of 2, also changing the number of blocks, from 1 to the maximum number of blocks of the problems, and the number of rectangles {1e7, 5e7, 1e8, 5e8}.

Show the speed-ups obtained and your conclusions.

4. Complete the file "mat\_vec\_row.cu", so that all threads work together to compute a matrix-vector product.

Each block of threads must manage a matrix block, whose number of columns is a multiple of the block of threads, and whose number of rows is variable. During the computation, a local subvector is computed which has to be accumulated on the result.

Execute the code in patan changing the number of threads, from 256 to 1024 using powers of 2, also changing the number of blocks from 1 to the maximum number of blocks of the problems, as well as the number of rows (M) and columns (N), whose values should be { 80, 800, 8000 }.

Show the speed-ups obtained and your conclusions.