Additional assignment

With reference to the problem scenario identified in the previous assignment, you shall implement a modified kernel that is capable of computing

$$Y \leftarrow AX,$$

where A is a sparse matrix stored in CSR or ELLPACK format, as in the base assignment, whereas X and Y are "tall" dense matrices, i.e. matrices with a limited number of columns. In particular, if M is the size of the matrix A, you shall consider X an Y of size $M \times 2$, $M \times 3$, $M \times 4$ and $M \times 8$. Notice that the number of floating point operations involved in the product of the sparse matrix A by an $M \times k$ matrix is $2 \times NZ \times k$.

The code shall be implemented in OpenMP and CUDA.

How does the performance (in MFLOPS) you obtain vary with the number of columns?

Note that in the original description of the assignment, the format for ELLPACK was described in terms of a 2D array with contiguous allocation; in MATLAB this is stored by columns, but in C-like languages the same mory layout is to be interpreted as a storage by rows with the dimension exchanged.

You shall deliver a report containing:

- A summary design specification;
- A summary test plan;
- Performance analysis;
- Source code.

The deadline for delivery is Jul 18th, 2017, at 1200 noon.