

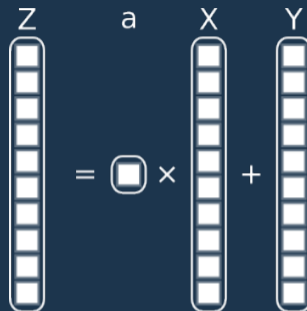
Optimisation : Saxpy

Pierre Aubert



The Single precision A X plus Y (saxpy)

$$z_i = a \cdot x_i + y_i, \quad \forall i \in 1, N, a \in \mathbb{R}$$

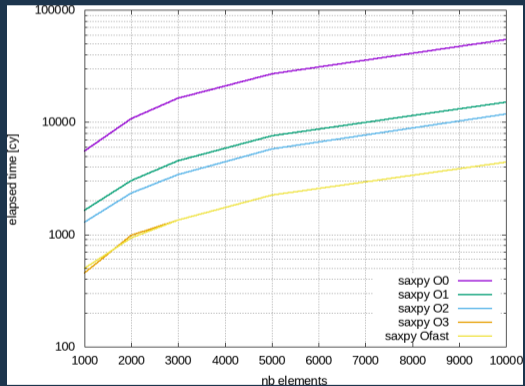


```
void saxpy(float* tabResult, float scal, const float * tabX, const float* tabY, long unsigned int nbElement){
    for(long unsigned int i(0lu); i < nbElement; ++i){
        tabResult[i] = scal*tabX[i] + tabY[i];
    }
}
```

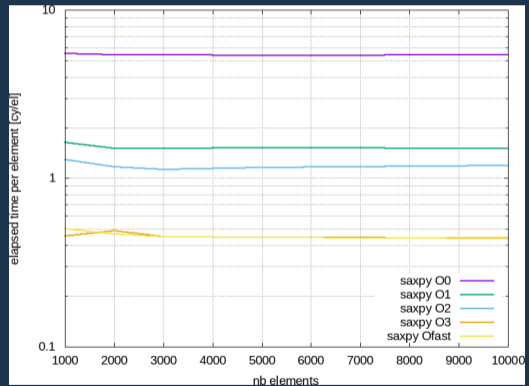
You already know how to make the performances test !

- ▶ Do not forget :
 - ▶ To include `asterics_hpc.h` in the `main.cpp`
 - ▶ To call `target_link_libraries(saxpy_O3 asterics_hpc)`

Total Elapsed Time (cy)



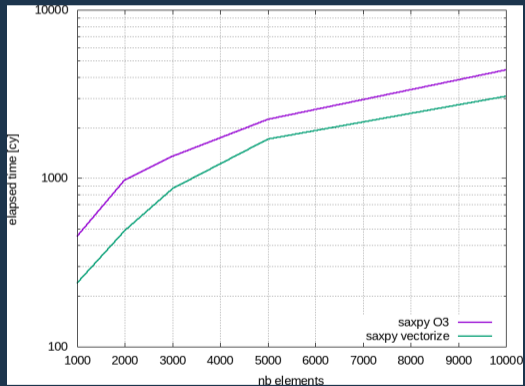
Elapsed Time per element (cy/el)



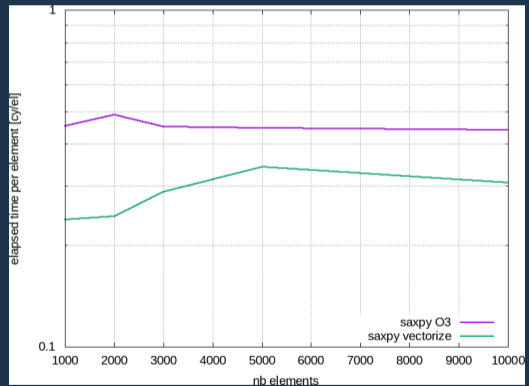
Speed up of 12 between -O0 and -O3 or -Ofast



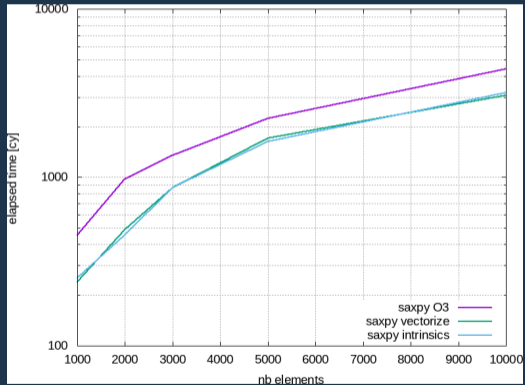
Total Elapsed Time (cy)



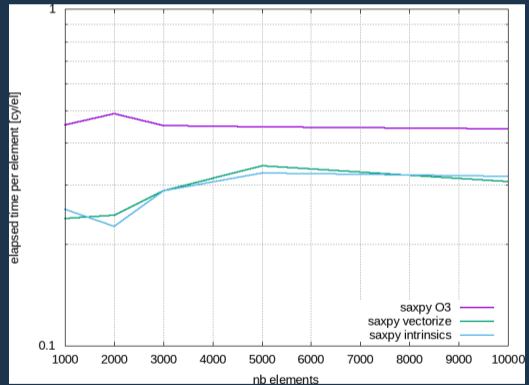
Elapsed Time per element (cy/el)



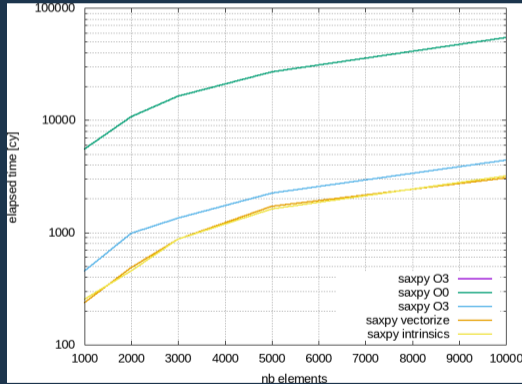
Total Elapsed Time (cy)



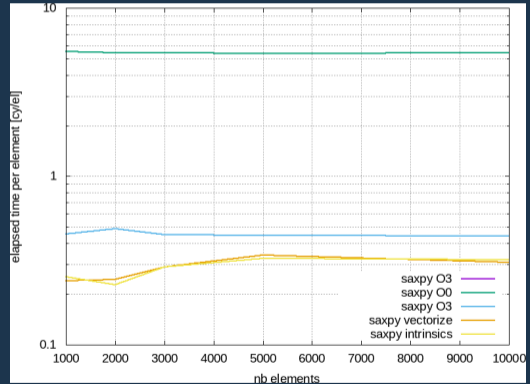
Elapsed Time per element (cy/el)



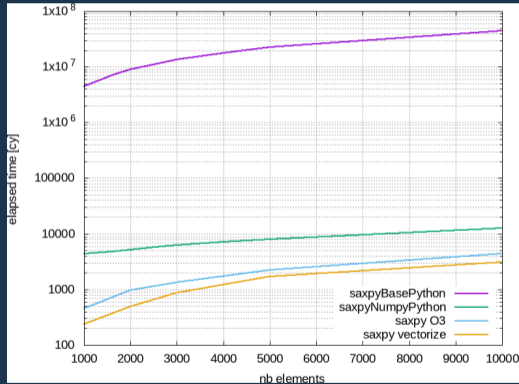
Total Elapsed Time (cy)



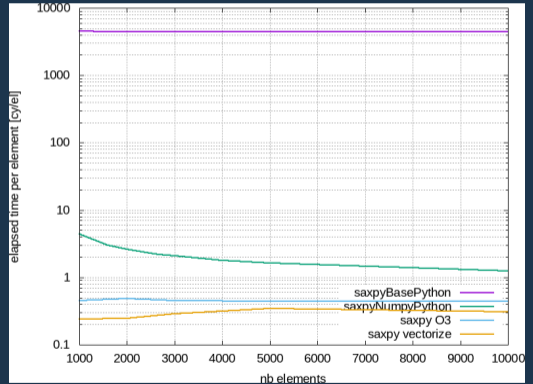
Elapsed Time per element (cy/el)



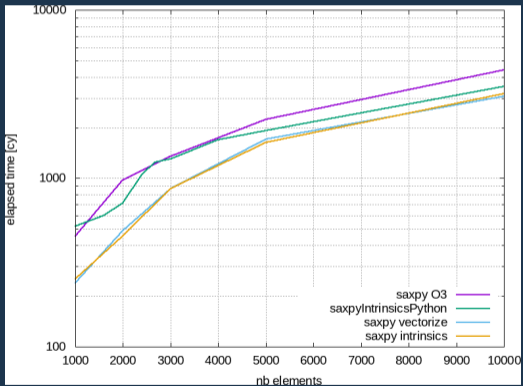
Total Elapsed Time (cy)



Elapsed Time per element (cy/el)



Total Elapsed Time (cy)



Elapsed Time per element (cy/el)

