

# *CPAR project 1*

## Performance evaluation of a single core

24/02/2016

---

In this project we will study the effect on the processor performance of the memory hierarchy when accessing large amounts of data. The product of two matrices will be used for this study.

1. Download the example file **product\_ex1.zip** that contains the basic algorithm in C/C++ that multiplies two matrices, i.e. multiplies one line of the first matrix by each column of the second matrix. Implement the same algorithm in another programming language, such as JAVA, C#, Fortran, etc.

Register the processing time for the several versions of the algorithm, for input matrices from 600x600 to 3000x3000 elements with increments in both dimensions of 400.

Use the Performance API (PAPI) to collect relevant performance indicators of the program execution.

2. Implement a version that multiplies an element from the first matrix by the correspondent line of the second matrix.

Register the processing time for the two versions of the algorithm, for input matrices from 600x600 to 3000x3000 elements with increments in both dimensions of 400.

Register the processing time from 4000x4000 to 10000x10000 with intervals of 2000.

Use the PAPI to collect relevant performance indicators of the program execution.

3. Implement a parallel version of the two versions of the algorithm in C/C++ using OpenMP. Register the processing time for input matrices from 600x600 to 3000x3000 elements with increments in both dimensions of 400 and using from 1 to 4 threads. Use PAPI to collect relevant performance indicators of the program execution.

### OUTCOMES

Write a report of up to 6 pages explaining the algorithm versions and analyzing the results obtained. Justify the performance parameters selected and use them to evaluate and compare the versions implemented.

**To be delivered on: 30/03/2016**

#### Parameters for Report Evaluation:

- Problem description and algorithms explanation;
- Performance metrics and evaluation methodology;
- Results and analysis;
- Conclusions;
- Writing quality.