

Project template (I)

- **Introduction.** An accurate description of the assigned project should be provided, including analysis of the sequential algorithm that solves the problem addressed in the project.
 - Pseudo-code, examples, graphs, figures, application instances etc. may be provided.
- **Parallel design.** Preliminary study about the opportunities for parallelism inherent in the problem sequential algorithm.
 - State of the art analysis should be performed on parallel design strategies
 - Alternative designs, related to different parallelization strategies, should be considered and discussed at this stage, appropriately motivating why some operations lend themselves to effective parallelization and others do not as well as why some data structures may or may not minimize the burden of communication and synchronization.
 - Reference to prior work as well as link with related work must be clearly stated in the report.
 - Hybrid parallelization strategies, with data dependencies must be discussed too

Project template (II)

- **Implementation.** C implementation (C++ is also allowed). The code must be properly commented. In case the student identifies multiple parallelization strategies, several implementations can be provided discussing pros and cons of each strategy. The report can include some code snippets related to the most critical and interesting parts.
 - A link to the repo must be provided too.
 - Hybrid parallelization is recommended though it is not mandatory
- **Performance and scalability analysis.** The student must analyze the performance of the developed implementation in terms of execution time, speedup, and efficiency.
 - Both strong scalability and weak scalability should be evaluated where possible.

Evaluation and significance of results

- The **evaluation** will also take into account:
 - Clarity and effectiveness of presentation;
 - Depth, correctness and originality of theoretical analysis;
 - Technical skills and documentation of implementation;
 - Number and quality of multiple parallel strategies, if any;
 - Critical thinking in performance evaluation and analysis;
- **Significance of results:** since parallel programming is primarily concerned with performance, a good project must also provide a significant speedup.
- Note: The report should be as concise as possible, but without detriment to clarity of presentation. Expected size: 10 pages.

Relevant aspects for your project

- Cite references regarding papers or other material you got inspired by (state of the art analysis is part of the report!)
- Cite data sources you are using for your project
- Finding input data is challenging? Create a simple random generator app to create synthetic data
 - Different datasets (small, medium, large size) and then compare the performance of your application on the 3 input data
- Replicating/inspired by an existing work? Some tips
 - Reproduce and compare the results. Comment differences and other findings
 - Start from pseudocode and provide your C implementation; compare with findings from the authors
 - Add from existing code and add your own delta; then compare
 - Take 2 different implementations and port them on the cluster; compare the two on the cluster + original work

Project and oral examination

- *Project:*
 - parallel application using MPI (and OpenMP)
 - OpenMP implementation is not mandatory (bonus)
 - to be developed in team (1-2 students)
 - will be discussed during the oral examination
 - All the team members must be present during the oral examination and the project presentation