Individual Project

1   The Problem

In this final mandatory task in the course, you will work on an algorithm of your choice. The assignment must involve implementation of the algorithm, optimization of the code and parallelization either with Pthreads or OpenMP.

In Appendix A, you will find a list of possible algorithms, please chose one of these.

2   The Report

The report should be written as a scientific paper of academic quality, written using either LaTeX (preferred) or a word processor (e.g. Word or LibreOffice), in either English (preferred) or Swedish. Plots can be created using e.g. Matlab or Python Matplotlib. The report should include the following:

1. *Introduction*, providing a background and motivation.

2. *Problem description*, presenting the task.

3. *Solution method*, description the algorithm, optimization and parallelization.

4. *Experiments*, presenting how you evaluate the performance of your solution along with your results and some observations and comments.

5. *Conclusions*, with explanations of the results and with ideas for possible optimizations or improvements.

6. *References*, listing relevant literature that was consulted in the project.

The code should be submitted as a compressed archive along with the report. Please submit the report itself as a PDF file.

3   Grading

The project will account for 60% of your final grade and the group project for 40%. When grading your work, we will take the following aspects into account:

- *Solution:* Choice of algorithm, serial and parallel efficiency.

- *Methodology:* Demonstration of correctness, and performance evaluation.

- *Code:* Design, robustness, documentation, and general quality.

- *Report:* Disposition, presentation of results, and language quality.

We will also take the complexity and difficulty of the problem into account when evaluating your work. It is also important that you meet the deadline for submission, late admission after the deadline will affect negatively on the grade.

A   Potential algorithms

- Quicksort, Mergesort, Bucketsort, or some other sorting algorithm
- MxM with Strassens algorithm
- Sparse Matrix-Vector multiplication
- LU-factorization (with or without pivoting)
- Solution of triangular systems (Lx=b or Ux=b)
- QR-factorization
- Power method for eigenvalues
- Conjugate gradient
- Game of life
- Numerical PDE solver