15. feb. 2019

Oblig 2 i IN3030 – v2019
Parallelization of Matrix Multiplication

This compulsory assignment (Oblig) is about Matrix Multiplication.

You are to implement and run three variants of Matrix Multiplication A x B: one using the classic algorithm, one where B is transposed, and one where A Is transposed. For each version, you should implement and run both a sequential and a parallel version of Matrix Multiplication and measure the resulting speedup.

You are to run with matrix sizes of 100x100, 200x200, 500x500, and 1000x1000.

Your program must be able to run regardless of the number of cores and should utilize them all.

In your solution include a check that you get the correct results, based on the sequential algorithm. Beware, that due to rounding errors, the results might differ very slightly.

Show the speedup results both in table form and show a graphical representation of the speedup results. Furthermore, show a graph comparing all six variant – use the variant that is the fastest (for large N) as a baseline, so that the others are depicted relative to the fastest.

You must use the published precode that we provide on the course web site to fill the two arrays to be multiplied.

Precode: https://github.uio.no/magnuesp/IN3030-v19

Be aware that the sequential code can be pretty slow, especially for larger matrices.

You must write a short report explaining how you did the parallelization, how you synchronized and why (or why not) this gives a speedup. Comment on why the speedup varies with different sizes of the matrices. Include the table and graphs in the report. It should also contain a short user's guide explaining how to execute your code.
The report is to be delivered in PDF format. Your Java code is to be uploaded as a single zip file or tar file.

Delivery: Deadline is Monday March 4th, 2019 at 23:59 noon CET – that is 23:59:00 local time in Oslo.

Deliver in devilry.

Template for report:

1. Introduction
2. Sequential Matrix Multiplication – *short description*
3. Parallel Matrix Multiplication – *how you did the parallelization*
4. Measurements – *includes discussion, tables, graphs*
5. User guide – *how to run your program*
6. Conclusion – *just a short summary*