- As announced in the introductory lecture in week 35, we will not follow the course book in the first few lectures.
- The slides that will be used in week 36 can be found via the Blackboard System. We will cover / covered "Single Processor Machines: Memory Hierarchies and Processor Features" with the "Case Study: Tuning Matrix Multiply" in week 35 / week 36. The slides are the main source of information and should be sufficient for understanding. However, many articles are cited on the slides in case you want to get more information.
- Chapter 3 of the book "Dongarra, J., et al. 2002. The Sourcebook of Parallel Computing, Morgan Kaufmann." has been uploaded in the Blackboard system as additional reading material.
- The preliminary version of the first mandatory assignment is already online. End of this week the first mandatory homework assignment will be finalized.
- In week 36 we will start with "Introduction to Parallel Machines and Programming Models". The slides will be available very soon.

Exercises

Parallel Computing, DM818 (Fall 2011)

Department of Mathematics and Computer Science
University of Southern Denmark
Daniel Merkle

In the lectures in week 36 we discussed two different approaches for matrix-matrix multiplication, namely the naive approach and the blocked matrix-matrix multiplication. For both of them we determined the computational intensity. The goal of this exercise is to investigate a striped version of matrix-matrix multiplication. Suppose for the product $A = B \times C$ the matrices A, B, and C (each matrix has size $n \times n$) are striped in N stripes, i.e. matrices A and C are divided in N blocks of size $n/N \times n$, and matrix B is divided in stripes of size $n \times n/N$. Let $b = n/N$ denote the block size.

$$\begin{pmatrix} C_1 \\ \hline C_2 \\ \hline \dots \\ \hline C_N \end{pmatrix} = \begin{pmatrix} A_1 \\ \hline A_2 \\ \hline \dots \\ \hline A_N \end{pmatrix} \cdot \left(\begin{array}{c|c|c|c} B_1 & B_2 & \dots & B_N \\ & & & \end{array}\right)$$

a) Write the pseudo code for this algorithm.

b) Determine the number of slow memory accesses (suppose that three blocks of size $n \times b$ fit into fast memory).

c) What is the number of arithmetic operations?

d) Determine the computational intensity.

e) Compare this computational intensity with the computational intensity of the blocked version presented in the lecture.