

Program for multiplying matrices (sequential, omp, pthread)

Student: Benno Waldhauer

Table of contents

Requirements.....	1
Installation.....	1
Use of the program.....	2
Run a lot of test cases in one step.....	2
Logfile.....	3
Comparison of pthread and omp method.....	3

Requirements

- calculation of $A[n][n] * B[n][n] = C[n][n]$
- different sizes of matrices
- dynamical bufferallocation
- time measurement
- logging
- different methods for multiplication (sequential and parallel with openMP and PThread)

Installation

- unpack the benno_waldhauer_seq_omp_pthread.zip
- copy all files to your preferred folder
- go to these folder
- type in „make“ to compile the program

Use of the program

To start the program go to the installation folder.

There are 2 possibilities to start the program.

1. without passing arguments
 1. type `./prog`
 2. The program welcomes you and ask for the matrices dimensions. Type them in.
 3. Then you will be asked for the print options. Type in 0,1,2 or 3
 4. Then you will be asked for the comparison method. Type in 0 or 1
 5. The program calculates and compares and prints the results, every execution is logged (see log.txt)
2. with passing arguments
 1. type `./prog (int dimension n) (int dimen. m) (int dim. p) (int print option) (int method)`
e.g. `./prog 500 400 300 1 1` ($A[p][n] * B[n][p] = C[m][p]$)
e.g. `./prog 100 100 100 0 1` (print option: 0 nothing, 1 screen, 2 file, 3 screen and file, comparison method: 0 nothing, 1 compare with sequential results)
 2. The program calculates and compares and prints like in the first start possibility

Run a lot of test cases in one step

To execute a line of test cases execute `usecases.sh` by typing `./usecases.sh`. These script runs the program for a line of matrices dimensions. Every dimension size will be repeated 3 times with sequential and parallel (omp & pthread) methods. Every execution is logged (look at log.txt) but nothing will be printed to screen or file. The results will be not compared to a sequential version of multiplication. So larger matrices could be calculated. The script could be easily customized for different series of tests.

For test cases there will be used two square matrices ($A[n][n] * B[n][n] = C[n][n]$)

Logfile

In log.txt every execution will be logged.

The Header of log.txt

LogID;Timestamp when Programm starts [s from 01.01.1970];
m,n,p;Dimensions of the matrices ($A[m][n] * B[n][p] = C[m][p]$);
print option;0=nothing 1=screen 2=file(LogID_XXXXXXXXXX.txt) 3=screen and file;
method; Compare results of parallel multiplications to 0 = nothing or 1 = seq. multiplication
multipliTime;Time[s] for multiplication;
multipliType;Type of multiplication;
numOfProc;Number of Processors on the machine;

LogID;print option;method;m;n;p;multipliTime;multipliType;numOfProc;

You can import the file log.txt with e.g. open office calc easily and work with the log data. (Have a look at ods_import.pdf)

If print option 2 or 3 were chosen, a file Result_LogID_XXXXXXXXXX.txt will be printed. These file also can be imported in calculation programs easily.

All log-files are located in the „log“ folder.

Comparison of pthread and omp method

For square matrices with a dimension lower than 1000 the omp method for multiplication seems to be faster than the parallel method with pthread. When the dimensions of the matrices to multiply become bigger than 1000 x 1000, then the pthread method becomes faster. But all in all the running time seems to be relatively similar. Also have a look at „ods-import.pdf“. There you will find a visual idea of this.