

README

The Documentation of Lab 1 Assignment

From: Chaolun Wang

Date: 09/06/2016

This folder includes 4 files/folders:

README.pdf: which is the documentation of the program

heated_plate.cpp: which is the c++ source code file for this lab assignment, the modify was done by Chaolun Wang in 09/06/2016

report.pdf: which is the report of the assignment, containing the execution time and the discussion of this lab

code&documentation_for_exercise: supplementary folder containing the exercise code and documentation.

To compile and run the code:

linux environment with g++ compiler is recommended. Open_MP library needs to be installed to the system. In linux terminal, firstly type:

g++ heated_plate.cpp -fopenmp

to compile the code and link with openmp library, after this command, the a.out should be generated. Then change the file name by the command:

mv a.out heated_plate

After that, the environment thread number is setted by the command:

export OMP_NUM_THREADS=1

The number 1 can be replaced by any preferred thread numbers. And the program is executed by entering:

./heated_plate 0.001

Test cases:

This program was runned three times with the thread number to be 1, 2 and 4. The programmed was runned on linux virtual machine installed on personal computer from Chaolun Wang. The virtual machine contain 2 processors and can run maximum 2 threads simultaneously.

Result:

Test case 1:

When executing with thread number setted as 1, I got following result:

HEATED_PLATE

C++ version

A program to solve for the steady state temperature distribution over a rectangular plate.

Spatial grid of 100 by 500 points.

The iteration will be repeated until the change is ≤ 0.001

Iteration Change

| | |
|------|-------------|
| 1 | 0.29264 |
| 2 | 0.218721 |
| 4 | 0.158882 |
| 8 | 0.113493 |
| 16 | 0.0802539 |
| 32 | 0.0563408 |
| 64 | 0.0392859 |
| 128 | 0.0271746 |
| 256 | 0.0185904 |
| 512 | 0.012354 |
| 1024 | 0.00696507 |
| 2048 | 0.00245445 |
| 3006 | 0.000999329 |

Error tolerance achieved.

HEATED_PLATE:

Normal end of execution.

The number of threads available:1

Elapsed wall clock time = 1.9358

Test case 2:

When changing the thread number to 2, I got the following result:

HEATED_PLATE

C++ version

A program to solve for the steady state temperature distribution over a rectangular plate.

Spatial grid of 100 by 500 points.

The iteration will be repeated until the change is ≤ 0.001

Iteration Change

| | |
|---|----------|
| 1 | 0.29264 |
| 2 | 0.218721 |

4 0.158882
8 0.113493
16 0.0802539
32 0.0563408
64 0.0392859
128 0.0271746
256 0.0185904
512 0.012354
1024 0.00696507
2048 0.00245445

3006 0.000999329

Error tolerance achieved.

HEATED_PLATE:

Normal end of execution.

The number of threads available:2

Elapsed wall clock time = 1.28991

Test case 3:

At last, when I changed the thread number to 4, I got the following result:

HEATED_PLATE

C++ version

A program to solve for the steady state temperature distribution
over a rectangular plate.

Spatial grid of 100 by 500 points.

The iteration will be repeated until the change is ≤ 0.001

Iteration Change

1 0.29264
2 0.218721
4 0.158882
8 0.113493
16 0.0802539
32 0.0563408
64 0.0392859
128 0.0271746
256 0.0185904
512 0.012354
1024 0.00696507
2048 0.00245445

3006 0.000999329

Error tolerance achieved.

HEATED_PLATE:

Normal end of execution.

The number of threads available:4

Elapsed wall clock time = 1.73073