CSC4005: Distributed and Parallel Computing

Assignment3: N-body Simulation

1 description

Write a sequential program, a P-thread program, an OpenMP program, and an MPI program to simulate an astronomical N-body system, but in two-dimensions. The bodies are initially at rest. Their initial positions and masses are to be selected randomly (using a random number generator). The gravity between N-body should be described by the following equation:
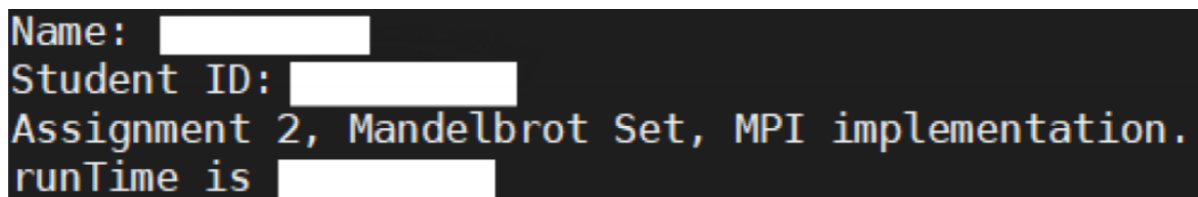
$$F = G\frac{m_1 \times m_2}{r^2} \tag{1}$$

Also you should consider the collision and bouncing, otherwise, all the points will be collasped into a singular point. Display the movement of the bodies using xlib or other GUI systems on your computers.

Bounus(10 points): Write an MPI + OpenMP program for a two-dimensional N-body simulation.(Bounus will be added to the assignment's score. The maximum score is 110, and you need to hand in the code file for it and should conduct a simple running time analysis in your report).

2 requirement

- You need to implement four versions of the tasks, which are **Sequential version**,**openMP version**,**MPI version** and a **Pthread version**. And hand in the codes for these four versions in four seperate code files. You need to print the following information for your codes including your **name, student id, assignment id, implementation version, running time of the program**.(see the folloing figure.)

- In your submit code, it should display an image with size of 200 × 200, and with 200 bodies moving for a fixed number of iterations.

- Include the results in your report by capturing a picture on your screen.

- Record a video of your result.

- For the openMP program, you should evaluate it like Pthread with specific number of threads.

- You need to specify the **command line** about how to **compile** and **run** your program.

- You need to **compare the performance** of different implementation and configurations in your report.

- The number of processes or threads used in the program (<u>up to **at least** 16 processes and threads</u>)
- MPI vs Sequential vs Pthread vs OpenMP
- <u>you should fix the size of the display window and the number of iterations, instead you should change the number of bodies (ranging from small, medium to large size).</u>
- More if you have

- You need to include **three figures** describing the structure of your MPI program, Pthread program and openMP program.

- The report should be written in appropriate format which you could refer to the report template.

3 Tips

- you do not need to refresh your screen every iteration, you can refresh it <u>every fixed interval of iterations.</u>

- Finish the sequential version firstly. Only then start to finish the other versions.

- When measuring the running time of the progra<u>m, make sure only measure the computation time, and with communication time if using MPI</u>. (Because Xlib's drawing is time consuming.)

- You should start your homework as soon as possible, do not try to finish it in the last two days before the deadline.

- Debug your program on the VM that is built on the image ubuntu.vdi. Make sure the program is OK, only after then measure the running time of the program on the server.

- Make sure use 'qsub' command to submit your jobs onto the master machine, do not directly run your program on the master machine.

- Try to limit your program running time within 60 seconds. If your program is running slow, try to improve your code or reduce the image's size.

- The 8 VMs on the server is now split into two cluster (each cluster has 20 cores), you can choose one of them to submit your job. See job_submission.pdf for more details.

4 Where and What to Turn in Your Homework

- Please turn in a zip file includes
  - Report
  - Codes
  - video demonstrating your program results

- zip your source codes, result video, paper in a zip file, and name it studentID.zip, then submit it on Blackboard.

- Late submission penalty, 5 points deduction for each 12 hours after the deadline.

5 Due:

23:59,Nov,24,2020