High Performance Computing

Assignment-2

Report

Team No: 9

Team Members:

Anirudh Chimpidi – SE20UCSE019

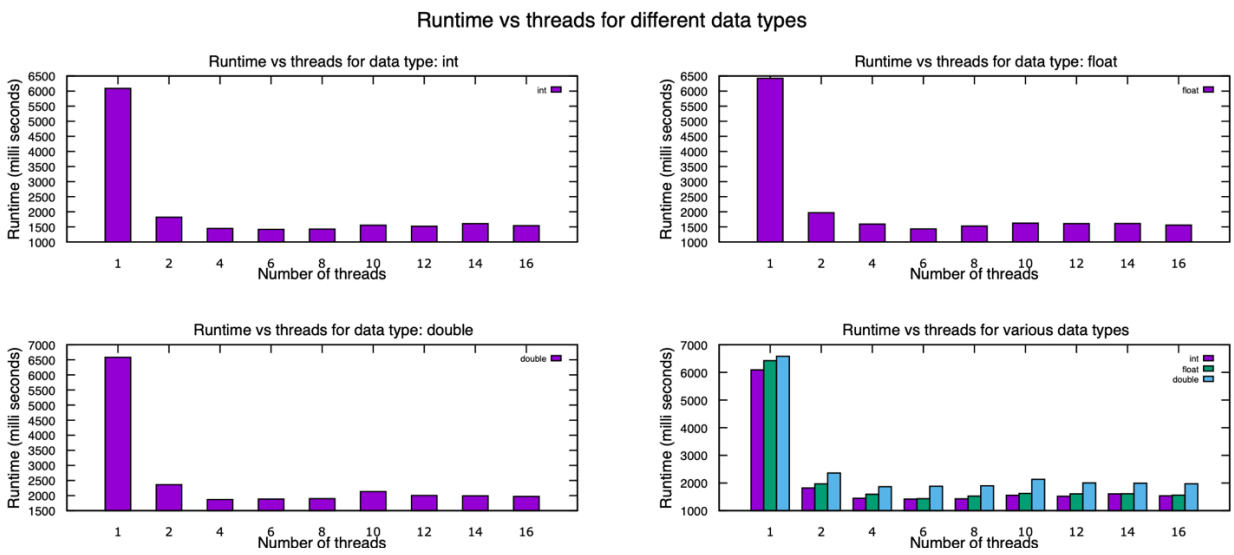
Naga Tharun Makkena – SE20UCSE105

Sri Harsha Vandanapu - SE20UCSE184

Rohan Potta - SE20UCSE145

Dhanush Bommavaram - SE20UCSE039

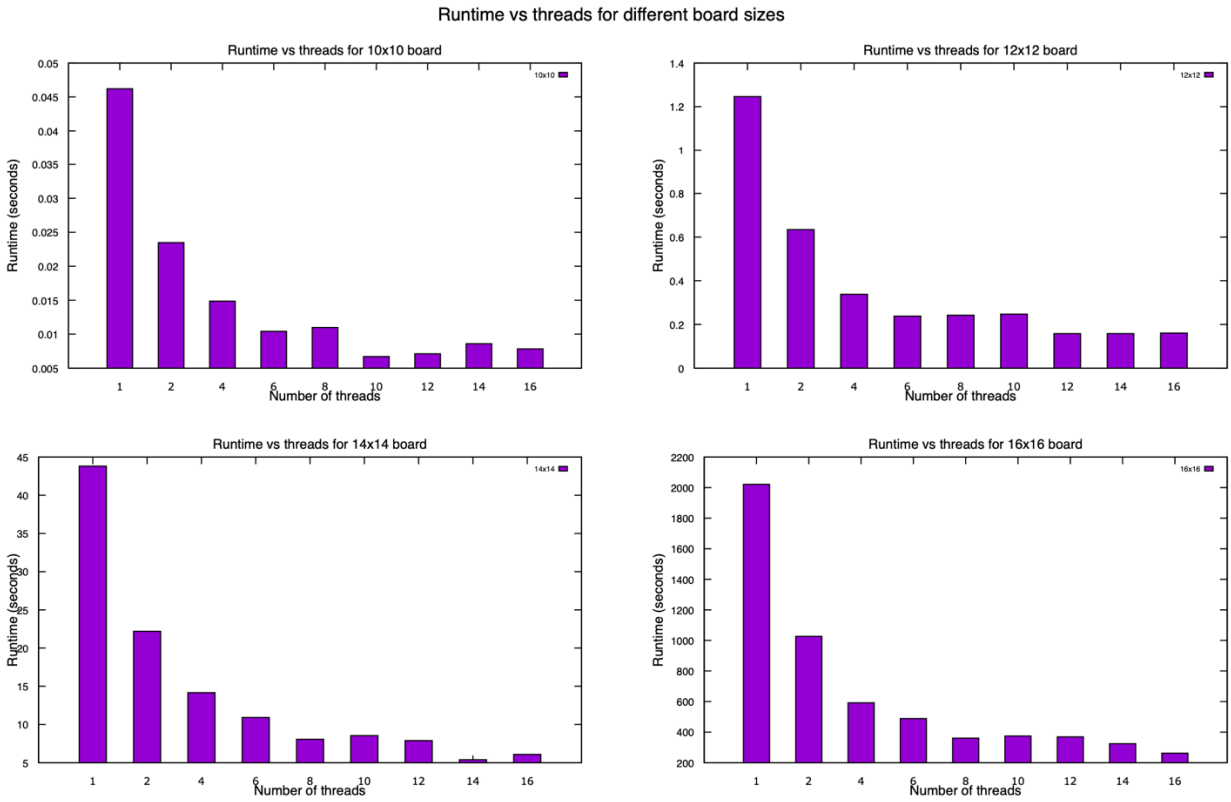
Q1) Merge Sort (Bottom up, In-place Method)



Here, we can observe that by keeping the size of the array constant but changing the number of threads, we can decrease the time exponentially. The run reduces significantly from 1 thread o 2 threads, the speed up is almost 10 times that of single thread execution.

From the fourth graph, we can compare the execution times for different data types, the double takes more time than the other types. The decreasing order of the execution times is double, float, int.

Q2) N-Queen Problem



Here we have the graphs varying the board size from n = 10, 12, 14, 16. The number of solutions present for each board is 724, 14,200, 365,596, 14,772,512, respectively.

As we can see, the time taken to solve each question increases as we increase the size of the chessboard (size of n). However, the execution time is also reduced if we increase the number of threads, we use to solve the problem.

Naturally, a giant board will have more solutions therefore, they take longer to solve.