Operating Systems–2: CS3523
January 2022
Programming Assignment 1: Finding nearest point using multiple threads
Submission Date: 16th January 2022

Goal: Find the nearest point in a given set of points from a source in a two dimensional plane.

Details:

In this program we are implementing a simpler version of the Traveling Salesperson Problem. Given a source point (X,Y) and set of possible destinations { (X_1,Y_1), (X_2,Y_2), …… (X_n,Y_n)}. Find the nearest point to (X,Y) among the given set. Use the Euclidean distance formula for calculating the distance.

Euclidean Distance formula: (X,Y) to (X_1,Y_1) = $\sqrt{(X_1 - X\ )^2\ +\ (Y_1 - Y\ )^2)}$

Implementation:

Implement this program using threads. You are given as input *n* which is the total number of threads. The main thread creates *n* worker threads. Divide the set of destination points equally among *n* threads. Each thread finds the nearest point from the source among its sub-set. Then the thread terminates the closest point found to the main thread and then terminates. The main thread then determines the nearest point from the points given by the threads.

Input:

As input, the program will take the following in a file: (a) Number of threads (b) source point (X,Y) (c) Number of point in the set (d) Set of destination points { (X_1,Y_1), (X_2,Y_2), …… (X_n,Y_n)}

A sample input file is as follows:

```
2
(1,1)
4
(2,2) (4,5) (0,3) (1,2)
```

Output:

Your program should output the following: (i) Time taken to find the nearest point (ii) The nearest point coordinates

Sample output for the above input:

```
5 microseconds
(1,2)
```

Report Details:

As a part of this assignment, you have to prepare a report. The report will describe the low-level design of your program.

This report should contain a comparison of the performance of sequential and parallel execution in the form of graph plots. The comparison must consist of the following graphs:

1. Varying the number of threads: In this experiment you will vary the number of threads while keeping the total number of points fixed to 5000. The axes are as follows: (a) x-axis: number of threads varying as 1, 2, 4 and 8, 16. (b) y-axis: time taken to find the point nearest to a given source from the given set of 5000 points.

2. Varying the input set size: In this experiment, you will have to vary the number of points while keeping the total number of threads fixed to 16. (a) x-axis: the destination set sizes as 1000, 2000, 3000, 4000, 5000. (b) y-axis: time taken to find the point nearest to the given source point while having the total number of threads as 16.

Note that for execution with one thread, you can consider the normal sequential execution code.

Submission Format

You have to upload:
(1) The source code in the following format: Assgn1Src-<RollNo>.c

(2) Readme: Assgn1Readme-<RollNo>.txt, which contains the instructions for executing the program.

(3) Report: Assgn1Report-<RollNo>.pdf which will contain the report as described above.

Zip all the above documents. Name the zipped document as: Assgn1-<RollNo>.zip. Please follow this naming convention. Otherwise, your assignment will not be graded.

Please submit the zipped document on the classroom page by the above mentioned deadline.

Grading Policy:

The policy for grading this assignment will be -
(1) Design as described in report and analysis of the results: 50%;
(2) Execution of the tasks based on the description in the readme: 40%
(3) Code documentation and indentation: 10%.