

REPORT

Design:

The code starts with reading from a sample text file for the inputs. We must give the name of the input file. The input file contains the number of threads, source point, number of points in the set, and the list of points. By using this we will create the array of threads asked for using the 'pthread_t' type. Now we will create threads using pthread_create () and direct to 'doSlection ()' function to assign work to each thread to select the point from the part of array points given using the 'pthread_create ()' function in a for-loop. Before this, we will start the clock.

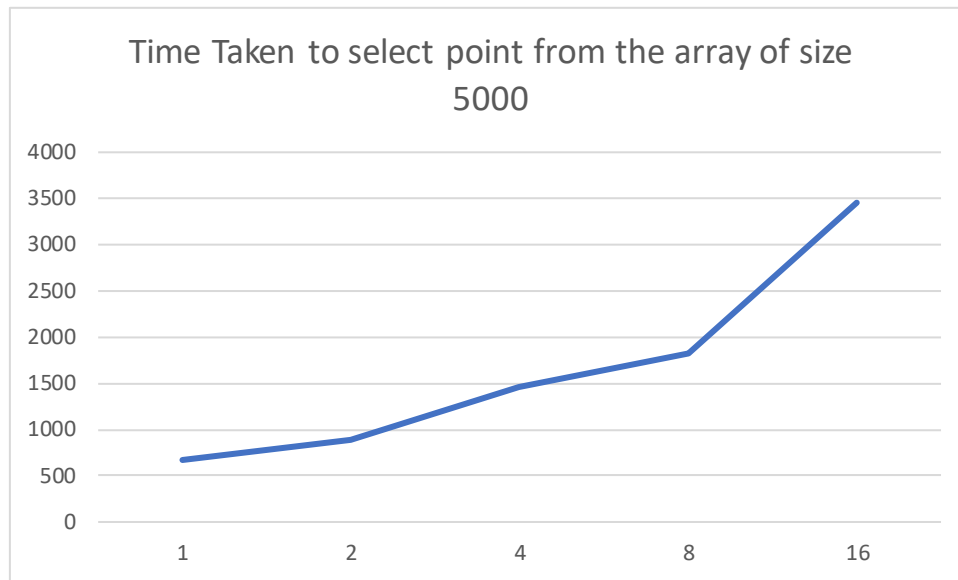
The 'doSelection ()' function divides the array points into parts equal to the number of threads and gives each part to each thread and then each thread does selection on a given part parallely and then ends the thread after completion. Now, we join back with the main thread using the 'pthread_join ()' function. We now merge all these parts, and we get the array of points selected from each part by each thread.

Comparison of performance:

(a) Varying the number of threads:

x-axis: number of threads varying as - 1, 2, 4, 8, and 16.

y-axis: time taken to select the point from random point array of size 5000.



(b) Varying the input array size:

x-axis: the array size varying as - 1000, 2000, 3000, 4000, 5000.

y-axis: time taken to select the point with a total number of threads as 16.

