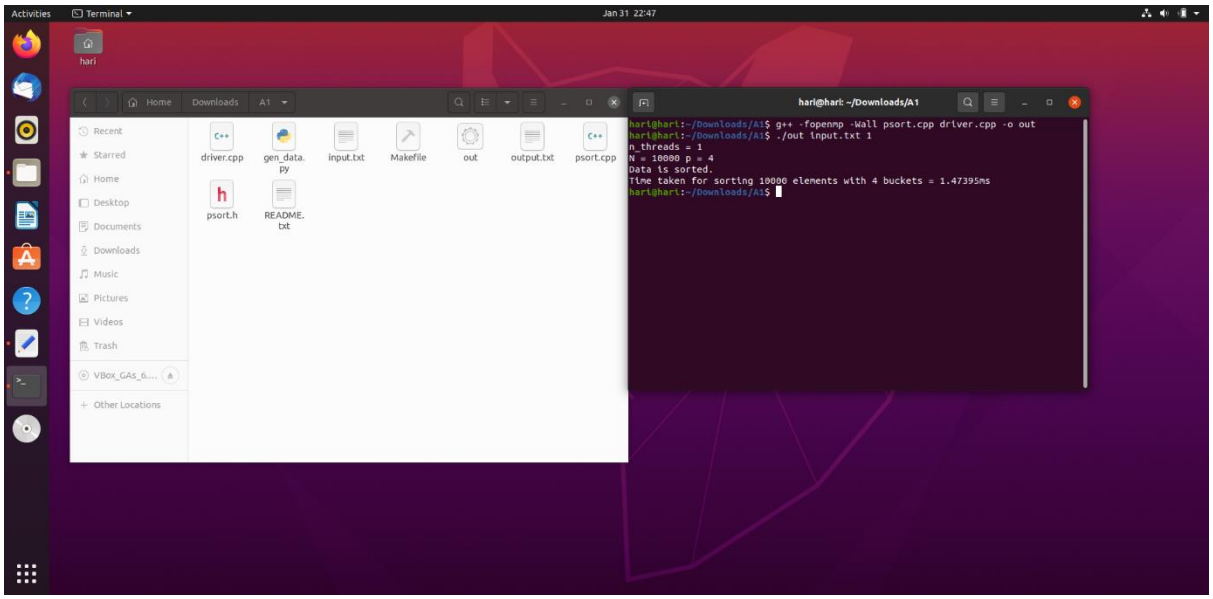
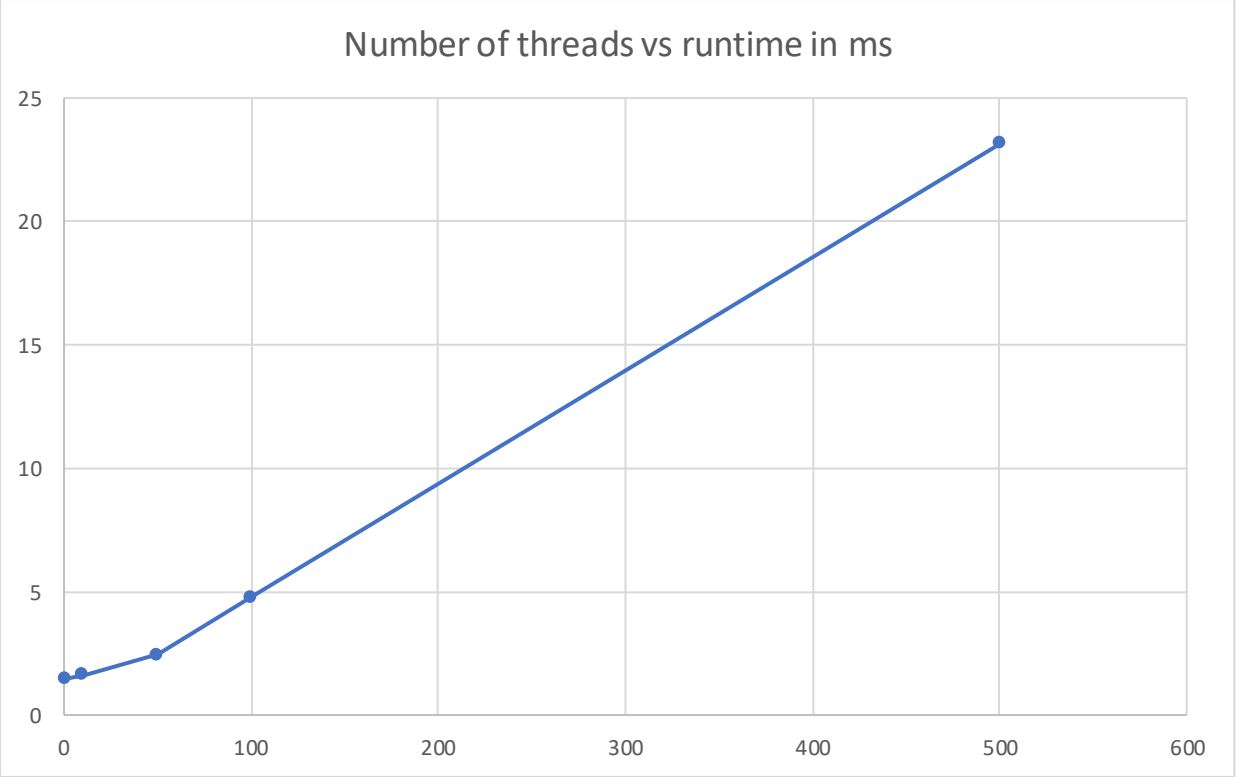
ASSIGNMENT 1 HARIKESH (2019CS10355)

Outline of the OPENMP task-based implementation and Design decisions: In case of the sequential sort I have used the quick sort algorithm and which choosing first of the p elements from the bucket I have assumed that if n/p is of the order of p and is actually less than p then I have just done sequential sort in that case. For the degree of parallelization of the code I have used the pragma task in case of distributing the values in different ranges and we have to wait for the task after that looping. And finally we have set the threshold to be 2*n/p and then checking whether the size in a specific bin is less than that or more than that and after that executed which of the sorting to be used parallel sorting or sequential sorting which to use.

I was not able to get the output from 12 CPUs and with array of size 2^24 on the HPC (took nealy 10 hours) and hence I have run it only on my local machine therefore submitting the graph obtained by the time of running on the local machine. Below is a snapshot showing the output of my file:



Now I can only draw the graph between the number of threads used and time taken by my local machine to run it. (as I was not able to run my code on 12 CPU's in HPC because of large amount of queuing time)


Number of threads vs runtime in ms
25
20
15
10
5
0
0
100
200
300
400
500
600