1. Pthread and OpenMP Based

- Write a ***MPI/pthread program*** for the following problem: "N thread are to be created, each taking numbers from the list to add to their partial sums. When all numbers have been taken, the threads can add their partial results to a shared location sum. A shared loacation global_index can be used by each thread to select next element of a[ ]."
- Design ***Parallel Vector Matrix Multiplication Algorithm*** and Implement using OpenMP using for and section directive; evaluate its performance for varying data size present results graphically using multiple treads.
- Design ***Parallel prime number generation Algorithm*** and Implement using OpenMP; evaluate its performance for varying data size and threads and present results graphically.
- Design ***Parallel Prefix Summation Algorithm*** and Implement using OpenMP; evaluate its performance for varying data size and threads and present results graphically.
- Design ***Parallel pi calculating Algorithm*** and Implement using OpenMP; evaluate its performance for varying data size and threads and present results graphically.
- Design a parallel C program which uses OpenMP to parallelize a simple example of ***Dijkstra's minimum distance algorithm*** for graphs, evaluate its performance for varying data size present results graphically using multiple treads.
- Design a parallel C program which demonstrates the computation of a ***Fast Fourier Transform*** in parallel, using OpenMP, evaluate its performance for varying data size present results graphically using multiple treads.
- Design a parallel C program which illustrates the use of the OpenMP application program interface to parallelize a ***Jacobi iteration solving A*x=b***, evaluate its performance for varying data size present results graphically using multiple treads.
- Design a parallel C program which ***produces an image of a Julia set***, using OpenMP to carry out the computation in parallel, evaluate its performance for varying data size present results graphically using multiple treads.
- Design a parallel C program which counts the number of ***primes between 1 and N***, using OpenMP for parallel execution, evaluate its performance for varying data size present results graphically using multiple treads.