## ISL77 - Parallel Programming Lab

**Part A: OpenMP Programs**

1. Write a program to sort an array on n elements using both sequential and parallel merge sort(using Section). Record the difference in execution time.
2. Estimate the value of pi using:

Parallelize the code by removing loop carried dependency and record both serial and parallel execution times.

1. Write an OpenMP program that divides the Iterations into chunks containing 2 iterations, respectively (OMP\_SCHEDULE=static,2). Its input should be the number of iterations, and its output should be which iterations of a parallelized for loop are executed by which thread. For example, if there are two threads and four iterations, the output might be the following:
   1. Thread 0 : Iterations 0 −− 1
   2. Thread 1 : Iterations 2 −− 3
2. Write a program to calculate n Fibonacci numbers using tasks.
3. Write a program to find the prime numbers from 1 to n employing parallel for directive. Record both serial and parallel execution times.
4. Parallel Vector Addition
5. Write a program calculate the sum of first 100 Numbers. (Using critical directive)

**Part B: MPI programs**

1. Demonstration of MPI\_Send and MPI\_Recv
2. Demonstration of deadlock using point to point communication
3. Avoidance of deadlock by altering the call sequence
4. Avoidance of deadlock by non-blocking calls
5. Avoidance of deadlock by using MPI\_Sendrecv (Bandwidth also)
6. Demonstration of synchronization between the two phases of program
7. Demonstration of Broadcast operation
8. Demonstration of MPI\_Gather
9. Demonstration of MPI\_Scatter
10. Demonstration of MPI\_Scatter and MPI\_Gather
11. Demonstration of MPI\_Reduce (MPI\_MAX, MPI\_MIN, MPI\_SUM, MPI\_PROD)
12. Demonstration of MPI\_Allreduce (MPI\_MAX, MPI\_MIN, MPI\_SUM, MPI\_PROD)

**Marks splitup:**

One program from each part.

Students have to execute 2 programs.

Part A Part B

Write up 4 4

Execution 20 15

Viva 7 (both part A and B)

Change of Program: -5 marks/part