Walchand College of Engineering, Sangli
Department of Computer Science and Engineering

Class: Final Year (Computer Science and Engineering)

Year: 2021-22 **Semester:** 1
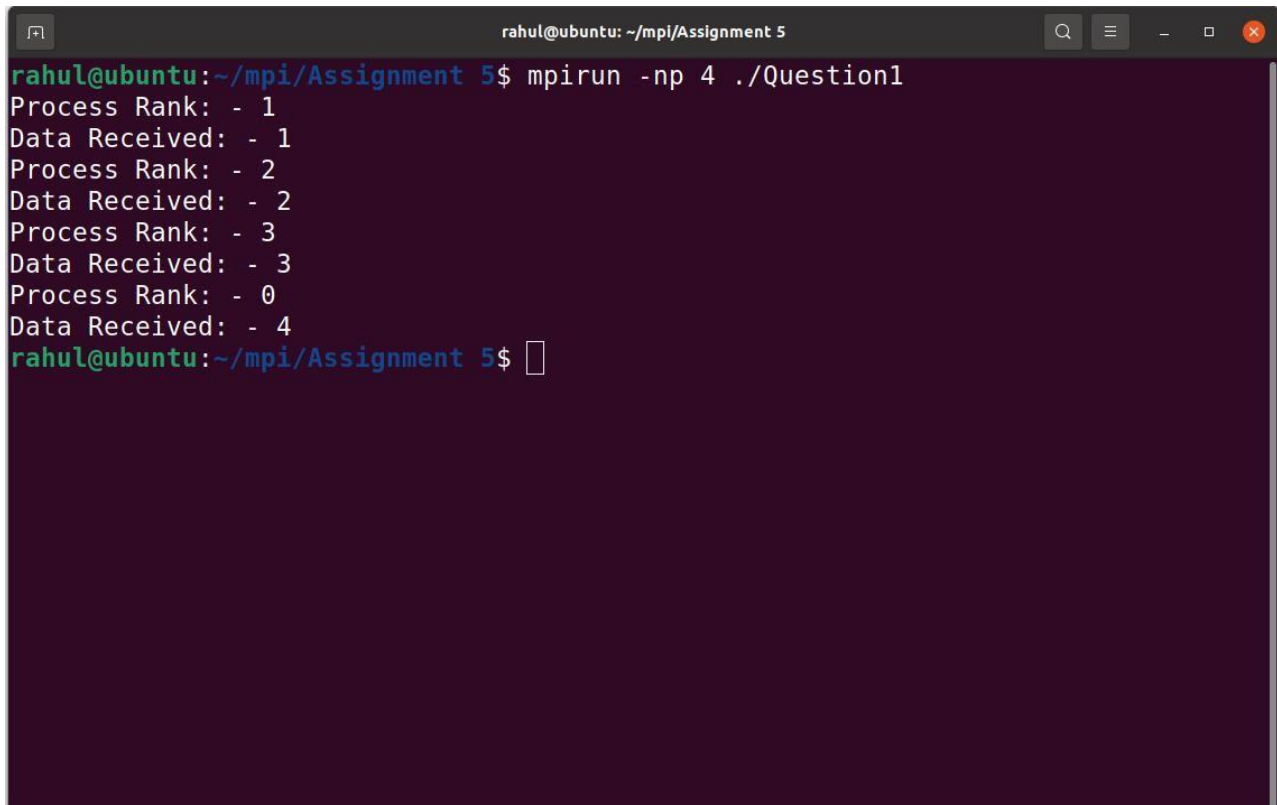
Course: High Performance Computing Lab

Practical No. 5

Exam Seat No: 2018BTECS00005

Name: Rahul Sanjay Naravadkar

Problem Statement 1: Implement a MPI program to give an example of blocking send and receive between four
processes.

Screenshot 1:

```
rahul@ubuntu:~/mpi/Assignment 5$ mpirun -np 4 ./Question1
Process Rank: - 1
Data Received: - 1
Process Rank: - 2
Data Received: - 2
Process Rank: - 3
Data Received: - 3
Process Rank: - 0
Data Received: - 4
rahul@ubuntu:~/mpi/Assignment 5$
```
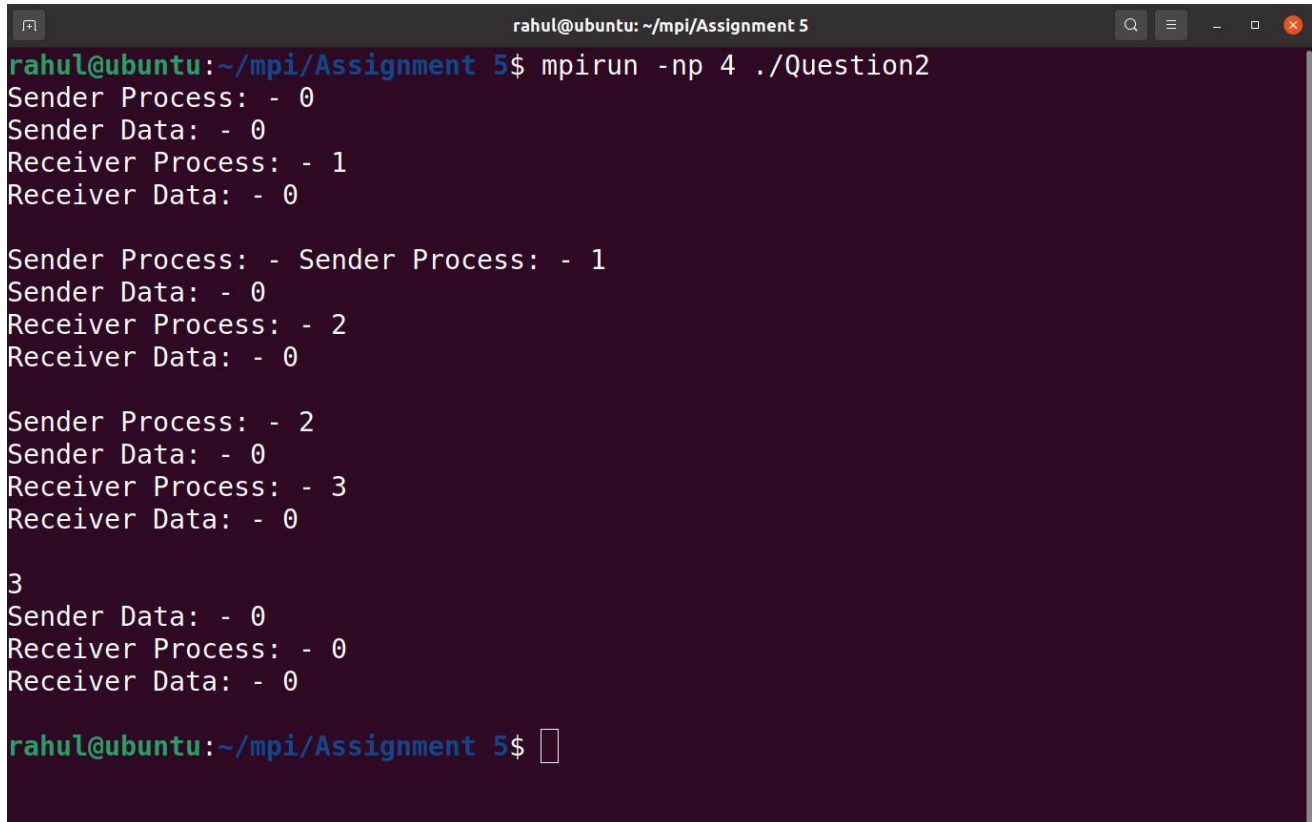
Information #:
Compile: - mpicxx -o Question1 Question1.cpp
Run - mpirun -np 4 ./Question1

Final Year: High Performance Computing Lab

Problem statement 2 : Implement MPI program using non-blocking send & receive functions to demonstrate. Nearest neighbour exchange of data in a ring topology

Screenshot:

```
                        rahul@ubuntu: ~/mpi/Assignment 5

rahul@ubuntu:~/mpi/Assignment 5$ mpirun -np 4 ./Question2
Sender Process: - 0
Sender Data: - 0
Receiver Process: - 1
Receiver Data: - 0

Sender Process: - Sender Process: - 1
Sender Data: - 0
Receiver Process: - 2
Receiver Data: - 0

Sender Process: - 2
Sender Data: - 0
Receiver Process: - 3
Receiver Data: - 0

3
Sender Data: - 0
Receiver Process: - 0
Receiver Data: - 0

rahul@ubuntu:~/mpi/Assignment 5$
```
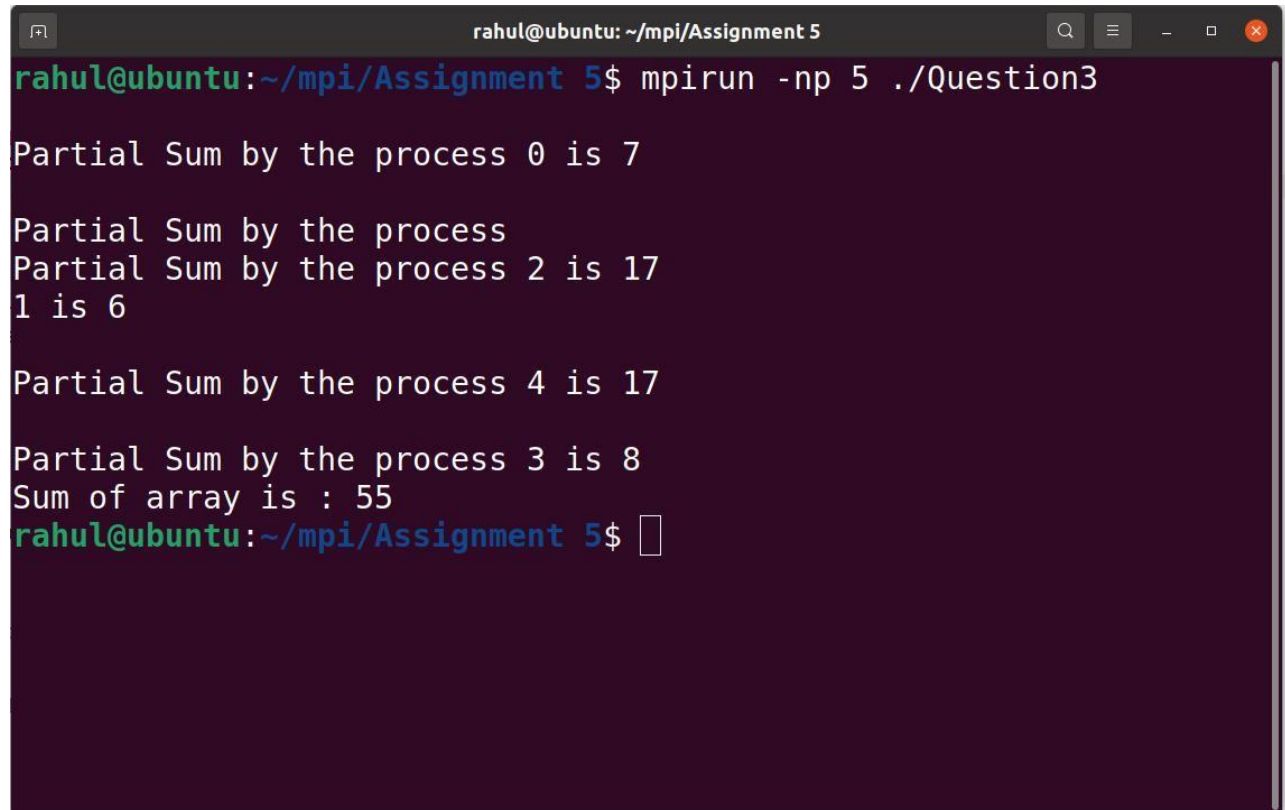
Information #:
Compile: - mpicxx -o Question2 Question2.cpp
Run - mpirun -np 4 ./Question2

Final Year: High Performance Computing Lab

Walchand College of Engineering, Sangli
Department of Computer Science and Engineering

Problem Statement 3:

Write a MPI program to find the sum of all the elements of an array A of size n using m number of processes. The two sums then are added to get the final result.

Screenshot:

Information #:

Compile: - mpicxx -o Question3 Question3.cpp

Run - mpirun -np 5 ./Question3

Github Link

Final Year: High Performance Computing Lab