

## Practical No. 5

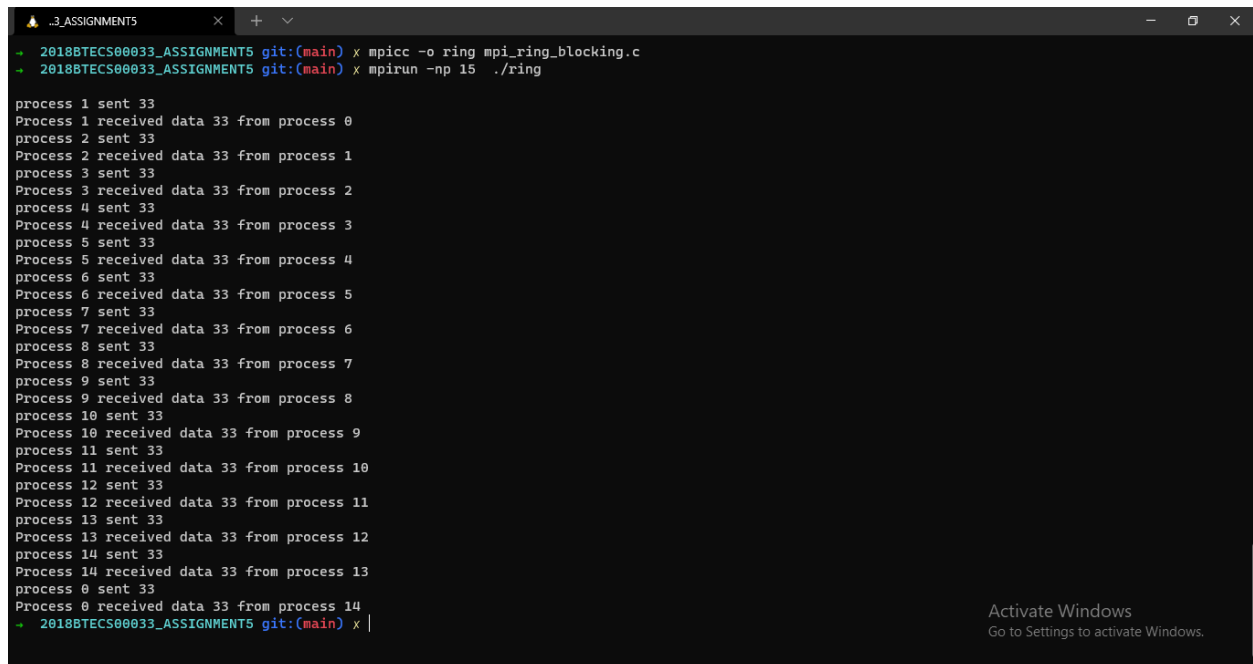
### Exam Seat No:

1. 2018BTECS00033 - Mahendra Bhimrao Garge

### Problem Statement 1:

Implement blocking and non-blocking MPI send & receive to demonstrate Nearest neighbour exchange of data in a ring topology.

### Screenshot 1:



```
.3_ASSIGNMENT5
2018BTECS00033_ASSIGNMENT5 git:(main) x mpicc -o ring mpi_ring_blocking.c
2018BTECS00033_ASSIGNMENT5 git:(main) x mpirun -np 15 ./ring

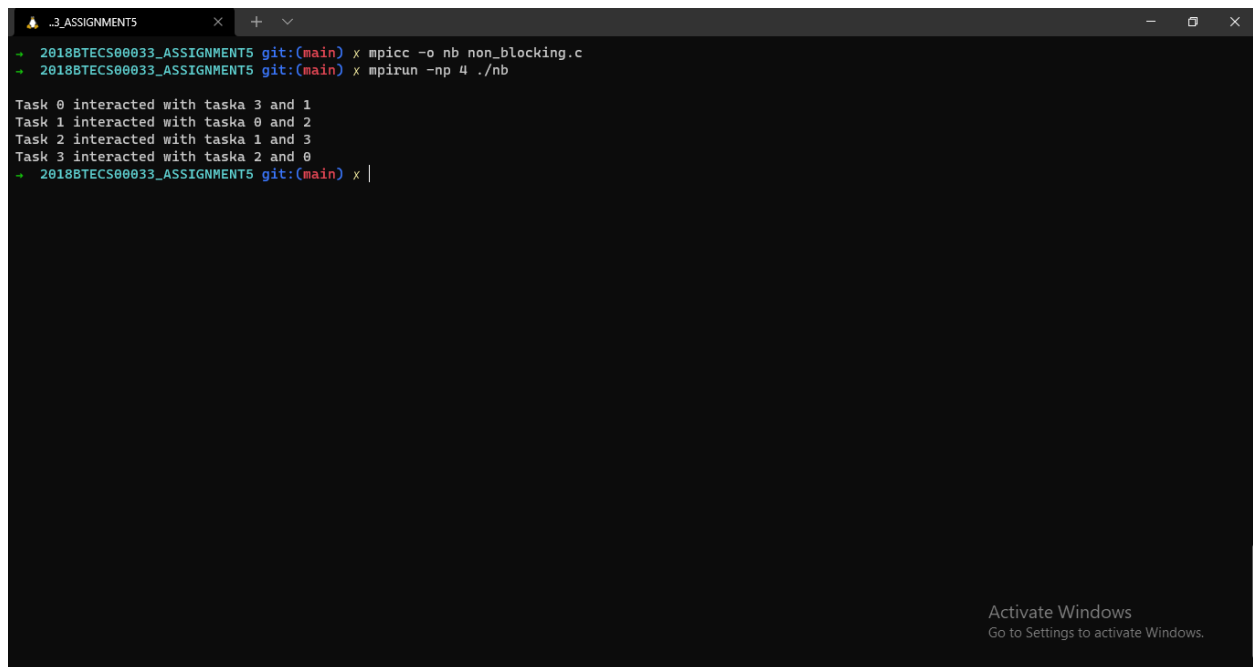
process 1 sent 33
Process 1 received data 33 from process 0
process 2 sent 33
Process 2 received data 33 from process 1
process 3 sent 33
Process 3 received data 33 from process 2
process 4 sent 33
Process 4 received data 33 from process 3
process 5 sent 33
Process 5 received data 33 from process 4
process 6 sent 33
Process 6 received data 33 from process 5
process 7 sent 33
Process 7 received data 33 from process 6
process 8 sent 33
Process 8 received data 33 from process 7
process 9 sent 33
Process 9 received data 33 from process 8
process 10 sent 33
Process 10 received data 33 from process 9
process 11 sent 33
Process 11 received data 33 from process 10
process 12 sent 33
Process 12 received data 33 from process 11
process 13 sent 33
Process 13 received data 33 from process 12
process 14 sent 33
Process 14 received data 33 from process 13
process 0 sent 33
Process 0 received data 33 from process 14
2018BTECS00033_ASSIGNMENT5 git:(main) x |
```

**Information 1:** Ring topology blocking send and receive with 15 processes.

### Problem Statement 1.2:

Implement a MPI program to give an example of non-blocking send and receive in ring topology.

### Screenshot 2:

A screenshot of a terminal window titled ".3\_ASSIGNMENTS". The terminal shows the compilation and execution of an MPI program. The commands entered are: `mpicc -o nb non_blocking.c` and `mpirun -np 4 ./nb`. The output shows four tasks interacting in a ring topology: Task 0 interacts with tasks 3 and 1; Task 1 interacts with tasks 0 and 2; Task 2 interacts with tasks 1 and 3; Task 3 interacts with tasks 2 and 0. The terminal window has a dark background and standard window controls at the top. An "Activate Windows" watermark is visible in the bottom right corner.

```
.3_ASSIGNMENTS
2018BTECS00033_ASSIGNMENTS5 git:(main) x mpicc -o nb non_blocking.c
2018BTECS00033_ASSIGNMENTS5 git:(main) x mpirun -np 4 ./nb

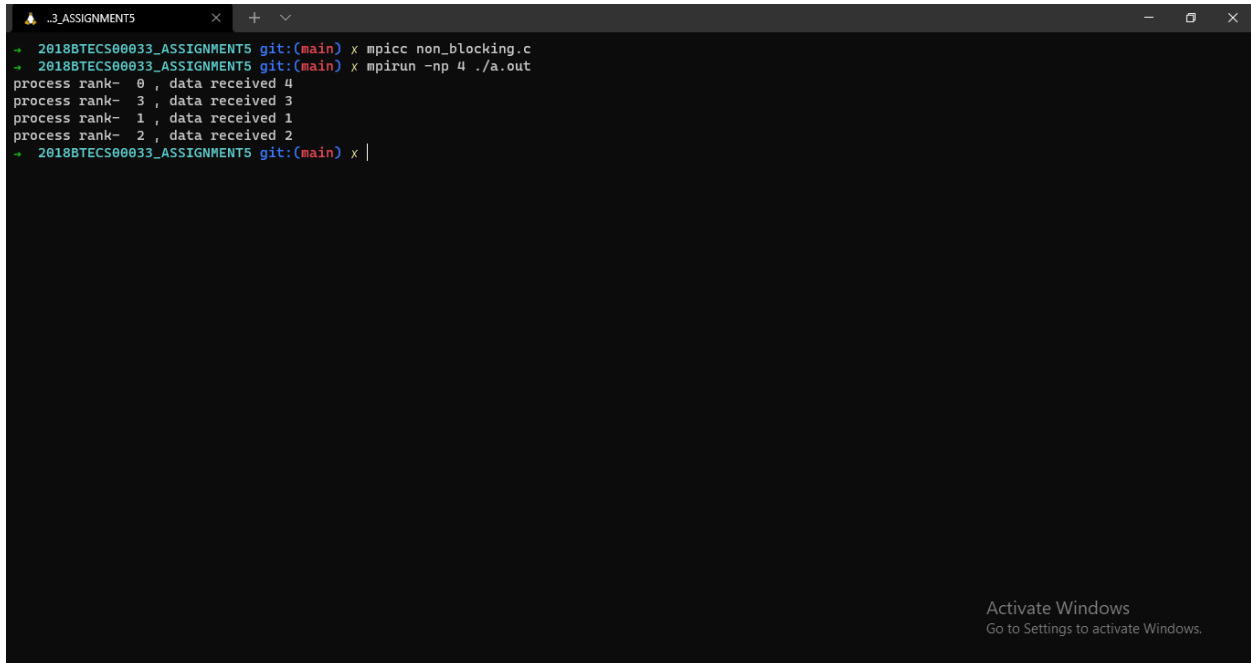
Task 0 interacted with taska 3 and 1
Task 1 interacted with taska 0 and 2
Task 2 interacted with taska 1 and 3
Task 3 interacted with taska 2 and 0
2018BTECS00033_ASSIGNMENTS5 git:(main) x |
```

**Information 2: Non-blocking send and receive between ring topology processes.**

### Problem Statement 2:

Implement a MPI program to give an example of non-blocking send and receive between four processes.

### Screenshot 3:



```
.3_ASSIGNMENTS
2018TECS00033_ASSIGNMENT5 git:(main) x mpicc non_blocking.c
2018TECS00033_ASSIGNMENT5 git:(main) x mpirun -np 4 ./a.out
process rank- 0 , data received 4
process rank- 3 , data received 3
process rank- 1 , data received 1
process rank- 2 , data received 2
2018TECS00033_ASSIGNMENT5 git:(main) x |
```

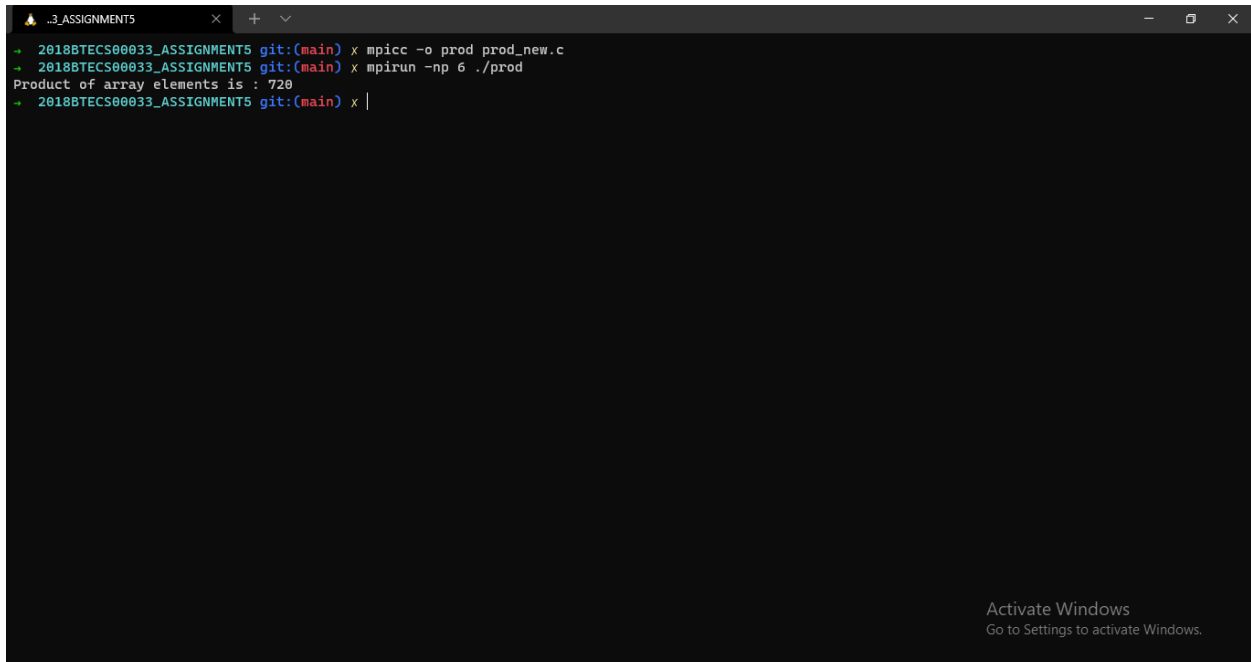
Activate Windows  
Go to Settings to activate Windows.

**Information 3: Non-blocking send and receive between 4 processes.**

### **Problem Statement 3:**

Write a MPI program to find the product 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 4:

A screenshot of a terminal window titled ".3\_ASSIGNMENTS". The terminal shows a series of commands and their outputs. The first command is "mpicc -o prod prod\_new.c", which is followed by "mpirun -np 6 ./prod". The output of the second command is "Product of array elements is : 720". The terminal window has a dark background with light-colored text. The window title bar shows standard Windows window controls (minimize, maximize, close) and a search icon. In the bottom right corner, there is a watermark that says "Activate Windows Go to Settings to activate Windows.".

```
.3_ASSIGNMENTS
> 2018TECS00033_ASSIGNMENTS5 git:(main) x mpicc -o prod prod_new.c
> 2018TECS00033_ASSIGNMENTS5 git:(main) x mpirun -np 6 ./prod
Product of array elements is : 720
> 2018TECS00033_ASSIGNMENTS5 git:(main) x |
```

#### Information 4: Product of array elements using MPI

Github Link: <https://github.com/g-mahendra/HPC LAB ASSIGNMENTS>