PDC - Lab 8

Aadhitya Swarnesh I

[redacted]

8 - October - 2020

Question 1

To demonstrate the use of MPI Barriers.

```
#include "mpi.h"
#include <stdio.h>
int main(int argc, char *argv[])
{
    int rank, nprocs;

    MPI_Init(&argc,&argv);
    MPI_Comm_size(MPI_COMM_WORLD,&nprocs);
    MPI_Comm_rank(MPI_COMM_WORLD,&rank);
    MPI_Barrier(MPI_COMM_WORLD);
    printf("Hello, world.  I am %d of %d\n", rank,
nprocs);fflush(stdout);
    MPI_Finalize();
    return 0;
}
```

```
(base) Aadhityas-MacBook-Air:8Oct2020 aadhitya$ mpicc p1.c
(base) Aadhityas-MacBook-Air:8Oct2020 aadhitya$ mpirun a.out
Hello, world.  I am 0 of 2
Hello, world.  I am 1 of 2
```

Question 2

To demonstrate the Token Ring implementation using MPI.

```
#include <mpi.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char** argv) {
  // Initialize the MPI environment
  MPI_Init(NULL, NULL);
  // Find out rank, size
  int world_rank;
  MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
  int world_size;
  MPI_Comm_size(MPI_COMM_WORLD, &world_size);

  int token;
  // Receive from the lower process and send to the
higher process. Take care
  // of the special case when you are the first process
to prevent deadlock.
  if (world_rank != 0) {
    MPI_Recv(&token, 1, MPI_INT, world_rank - 1, 0,
MPI_COMM_WORLD,
             MPI_STATUS_IGNORE);
    printf("Process %d received token %d from process
%d\n", world_rank, token,
           world_rank - 1);
  } else {
    // Set the token's value if you are process 0
    token = -1;
  }
  MPI_Send(&token, 1, MPI_INT, (world_rank + 1) %
world_size, 0,
           MPI_COMM_WORLD);
```

```
  // Now process 0 can receive from the last process.
This makes sure that at
  // least one MPI_Send is initialized before all
MPI_Recvs (again, to prevent
  // deadlock)
  if (world_rank == 0) {
    MPI_Recv(&token, 1, MPI_INT, world_size - 1, 0,
MPI_COMM_WORLD,
             MPI_STATUS_IGNORE);
    printf("Process %d received token %d from process
%d\n", world_rank, token,
           world_size - 1);
  }
  MPI_Finalize();
}
```