**Assignment\_1**

**Submitted by: Sehajleen kaur(102203342)**

**//mpiTrial**

#include <mpi.h>

#include <stdio.h>

int main(int argc, char\*\* argv) {

// Initialize the MPI environment

MPI\_Init(NULL, NULL);

// Get the number of processes

int world\_size;

MPI\_Comm\_size(MPI\_COMM\_WORLD, &world\_size);

// Get the rank of the process

int world\_rank;

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &world\_rank);

// Get the name of the processor

char processor\_name[MPI\_MAX\_PROCESSOR\_NAME];

int name\_len;

MPI\_Get\_processor\_name(processor\_name, &name\_len);

// Print off a hello world message

printf("Hello world from processor %s, rank %d out of %d processors\n",

processor\_name, world\_rank, world\_size);

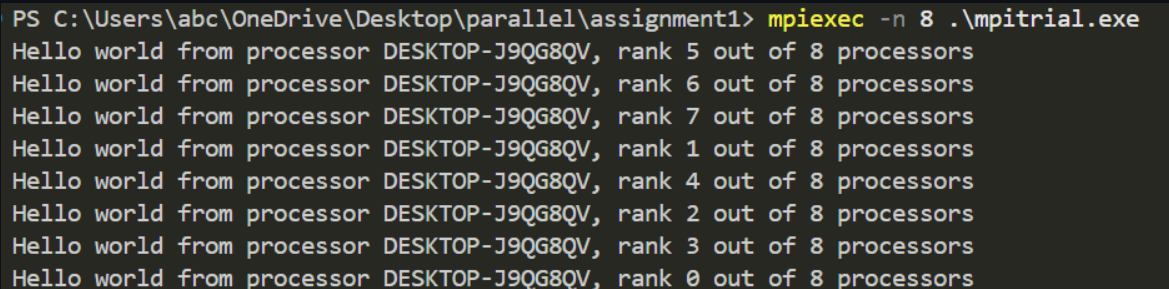
// Finalize the MPI environment.

MPI\_Finalize();

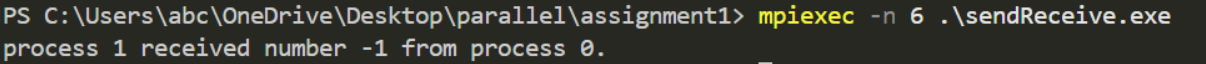
}

**//Outputs:**

MPI\_TRIAL(HELLO WORLD)



Send Receive



//SendReceiveCode

#include <mpi.h>

#include <stdio.h>

#include <stdlib.h>

//Basic Send functions in MPI

// MPI\_Send(

// void\* data,

// int count,

// MPI\_Datatype datatype,

// int destination,

// int tag,

// MPI\_Comm communicator

// )

//Basic Receive functions in MPI

// MPI\_Recv(

// void\* data,

// int count,

// MPI\_Datatype datatype,

// int source,

// int tag,

// MPI\_Comm communicator,

// MPI\_Status\* status

// )

int main(int argc, char\*\* argv){

MPI\_Init(NULL, NULL);

//rank

int world\_rank;

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &world\_rank);

//size

int world\_size;

MPI\_Comm\_size(MPI\_COMM\_WORLD, &world\_size);

// assuming 2 processors for this task

if(world\_size<5){

fprintf(stderr,"World size must be greater than 5 for %s.\n", argv[0]);

MPI\_Abort(MPI\_COMM\_WORLD, 5);

}

int number;

if(world\_rank==0){

number=-1;

MPI\_Send(

&number,

1,

MPI\_INT,

1,

0,

MPI\_COMM\_WORLD

);

}

else if(world\_rank==1){

MPI\_Recv(

&number,

1,

MPI\_INT,

0,

0,

MPI\_COMM\_WORLD,

MPI\_STATUS\_IGNORE

);

printf("process 1 received number %d from process 0.\n", number);

}

MPI\_Finalize();

}

//README\_FILE\_AS\_PER\_GITHUB:

