SL – V Exp 3:

Aim:

Design a distributed application using Message Passing Interface (MPI) for remote computation where client submits a string to the server and server returns the reverse of it to the client.

Steps:

open terminal

```
sudo apt-get update
sudo apt-get install libopenmpi-dev

mkdir exp3
cd exp3

gedit server.c
```

add following code in it

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

int main(int argc, char **argv)
{
  MPI_Comm client;
  MPI_Status status;
  char port_name[MPI_MAX_PORT_NAME],str[50],ch,temp;
  int size, again, i,j;

  MPI_Init(&argc, &argv);
  MPI_Comm_size(MPI_COMM_WORLD, &size);
  if (size != 1) {
    fprintf(stderr, "Server too big");
    exit(EXIT_FAILURE);
  }

  MPI_Open_port(MPI_INFO_NULL, port_name);
  printf("Server available at port: %s\n", port_name);

i=0;

 while (1) {
    MPI_Comm_accept(port_name, MPI_INFO_NULL, 0, MPI_COMM_WORLD, &client);
    again = 1;
    while (again) {
     MPI_Recv(&ch, 1, MPI_CHAR, MPI_ANY_SOURCE, MPI_ANY_TAG, client, &status);
      switch (status.MPI_TAG) {
```

```
    case 0:
        MPI_Comm_free(&client);
        MPI_Close_port(port_name);
        MPI_Finalize();
        return 0;

    case 1:
         printf("\nReceived String: %s\n",str);

        // reverse the string
              i = 0;
                 j = strlen(str) - 1;

                  while (i < j) {
                          temp = str[i];
                          str[i] = str[j];
                          str[j] = temp;
                          i++;
                          j--;
                  }

     printf("\nReversed string is : %s\n",str);

         // send the reversed string to client (character by character)

     for (i = 0; i < strlen(str); i++) {

          ch=str[i];

           MPI_Send(&ch, 1, MPI_CHAR, 0, 2, client);

      }

     //send tag=1 to indicate end of string
     MPI_Send(&ch, 1, MPI_CHAR, 0, 1, client);


     MPI_Comm_disconnect(&client);
        again = 0;
     strcpy(str,"");
     i=0;

         break;


    case 2:
        printf("Received character: %c\n", ch);

     str[i]=ch;
     i++;
      // add null character at the end of string
```

```
        str[i]='\0';
            break;
       default:
            /* Unexpected message type */
            MPI_Abort(MPI_COMM_WORLD, 1);
       }
     }
   }
}
```

save and exit the file

```
gedit client.c
```

add following code in it

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

int main( int argc, char **argv )
{
  MPI_Comm server;
  MPI_Status status;
  char port_name[MPI_MAX_PORT_NAME],str[50],ch;
  int i, tag,again;

  if (argc < 2) {
    fprintf(stderr, "server port name required.\n");
    exit(EXIT_FAILURE);
  }

  MPI_Init(&argc, &argv);
  strcpy(port_name, argv[1]);
  MPI_Comm_connect(port_name, MPI_INFO_NULL, 0, MPI_COMM_WORLD, &server);

  // accept input string
   printf("\nEnter the string :\n");
  scanf("%s",str);

  //send string to server (character by character)
  for (i = 0; i < strlen(str); i++) {

            if(str[i]!='\0')
               ch=str[i];

            tag=2;
            MPI_Send(&ch, 1, MPI_CHAR, 0, tag, server);

  }
```

```
// done sending string to the server
 MPI_Send(&i, 0, MPI_INT, 0, 1, server);

// Receive the reversed string from server and display it

  i=0;
  again=1;

   while (again) {
     MPI_Recv(&ch, 1, MPI_CHAR, MPI_ANY_SOURCE, MPI_ANY_TAG, server, &status);
      switch (status.MPI_TAG) {
      case 2:
                      str[i]=ch;
                i++;
                break;

      case 1:  again=0;
                break;

            }
 }

  printf("\nReversed string is : %s\n\n",str);


  MPI_Comm_disconnect(&server);
  MPI_Finalize();
  return 0;
}
```

save and exit the file

compile
mpicc server.c -o server
mpicc client.c -o client

run server
mpirun -np 1 ./server

it will display output similar to below (not necessarily the same)

Server available at port:
4290510848.0;tcp://192.168.1.101:35820;tcp://192.168.122.1:35820+4290510849.0;tcp://192.168.1.101:40208;tcp://192.168.122.1:40208:300

copy the **port-string** from the terminal output (e.g. the highlighted portion above)
we are going to supply this port-string as a first command line argument to the client

open another terminal
mpirun -np 1 ./client '4290510848.0;tcp://192.168.1.101:35820;tcp://192.168.122.1:35820+4290510849.0;tcp://192.168.1.101:40208;tcp://192.168.122.1:40208:300'

 # Don't forget to insert single quotes at the start & end of the port-string.

 # output : Server Terminal

```
s@mypc:~/e3$ mpicc server.c -o server
s@mypc:~/e3$ mpicc client.c -o client
s@mypc:~/e3$ mpirun -np 1 ./server
--------------------------------------------------------------------------
[[60663,1],0]: A high-performance Open MPI point-to-point messaging module
was unable to find any relevant network interfaces:

Module: OpenFabrics (openib)
  Host: mypc

Another transport will be used instead, although this may result in
lower performance.
--------------------------------------------------------------------------
Server available at port: 3975610368.0;tcp://192.168.1.101:50264;tcp://192.168.122
.1:50264+3975610369.0;tcp://192.168.1.101:39606;tcp://192.168.122.1:39606:300
Received character: c
Received character: o
Received character: l
Received character: l
Received character: e
Received character: g
Received character: e

Received String: college

Reversed string is : egelloc
```

Ouput: Client terminal

```
s@mypc:~/e3$ mpirun -np 1 ./client '3975610368.0;tcp://192.168.1.101:50264;tcp://1
92.168.122.1:50264+3975610369.0;tcp://192.168.1.101:39606;tcp://192.168.122.1:3960
6:300'
--------------------------------------------------------------------------
[[60271,1],0]: A high-performance Open MPI point-to-point messaging module
was unable to find any relevant network interfaces:

Module: OpenFabrics (openib)
  Host: mypc

Another transport will be used instead, although this may result in
lower performance.
--------------------------------------------------------------------------

Enter the string :
college

Reversed string is : egelloc

s@mypc:~/e3$
```
