Experiment No: - Sample

**Objective: - Write a program for addition of two numbers using RPCGEN.**

**add.x**

struct intpair

{

int a;

int b;

};

program PROG

{

version VERS

{

int ADD(intpair) = 1;

} = 1;

} = 234;

**add\_client.c**

/\*

\* This is sample code generated by rpcgen.

\* These are only templates and you can use them

\* as a guideline for developing your own functions.

\*/

#include "add.h"

void prog\_1(char \*host)

{

CLIENT \*clnt;

int \*result\_1;

intpair add\_1\_arg;

#ifndef DEBUG

clnt = clnt\_create (host, PROG, VERS, "udp");

if (clnt == NULL)

{

clnt\_pcreateerror (host);

exit (1);

}

#endif /\* DEBUG \*/

printf("\nName: Abhijit Parmar\n");

printf("\nEnrollment No:130380902015\n");

printf(" Enter the value a=");

scanf("%d",&add\_1\_arg.a);

printf(" Enter the value b=");

scanf("%d",&add\_1\_arg.b);

result\_1 = add\_1(&add\_1\_arg, clnt);

if (result\_1 == (int \*) NULL)

{

clnt\_perror (clnt, "call failed");

}

printf("\nAnswer=%d\\n",\*result\_1);

#ifndef DEBUG

clnt\_destroy (clnt);

#endif /\* DEBUG \*/server

}

int

main (int argc, char \*argv[])

{

char \*host;

if (argc < 2)

{

printf ("usage: %s server\_host\n", argv[0]);

exit (1);

}

host = argv[1];

prog\_1 (host);

exit (0);

}

**add\_server.c**

/\*

\* This is sample code generated by rpcgen.

\* These are only templates and you can use them

\* as a guideline for developing your own functions.

\*/

#include "add.h"

int \*

add\_1\_svc(intpair \*argp, struct svc\_req \*rqstp)

{

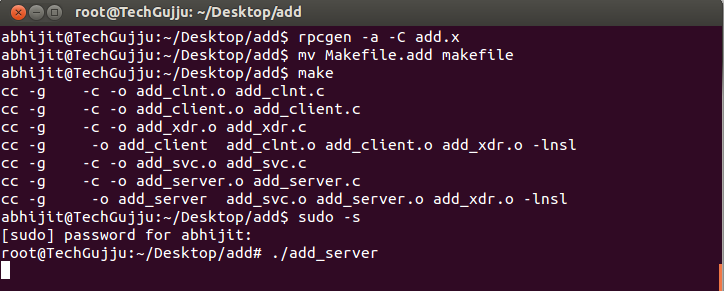
static int result;

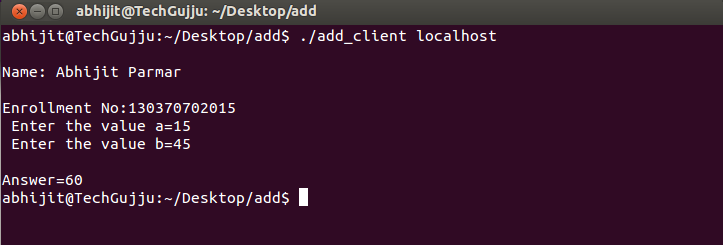
result = argp->a + argp->b;

return &result;

}

**Output:-**





131123131003

parth

131123131003

parth

Experiment No: - Sample

**Objective: - Write a program to implement Cristian algorithm using RMI.**

**cris.java**

import java.rmi.\*;

public interface cris extends Remote

{

public String time() throws RemoteException;

}

**client.java**

import java.rmi.\*;

import java.rmi.registry.\*;

import java.util.\*;

public class client

{

public static void main(String args[])throws Exception

{

Registry r = LocateRegistry.getRegistry();

cris c = (cris) r.lookup("time");

String tm = c.time();

System.out.println("Name:Abhijit Parmar\nEnrollment No:130370702015");

System.out.println(tm);

}

}

**server.java**

import java.rmi.\*;

import java.rmi.server.\*;

import java.rmi.registry.\*;

import java.util.\*;

public class server implements cris

{

String time="";

public String time()

{

Date d = new Date();

return d.toString();

}

public static void main(String args[])throws Exception

{

cris demo = new server();

cris stub = (cris)UnicastRemoteObject.exportObject(demo,0);

Registry r = LocateRegistry.getRegistry();

r.rebind("time",stub);

System.out.println("server started...:)");

}

}

**Output:-**

