NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR

Cachar, Assam

B.Tech. Vth Sem

Subject Code: CS-311

Subject Name: Computer Network Laboratory

Submitted By:

Name : Subhojit Ghimire

Sch. Id. : 1912160

Branch : CSE – B

- Q.5. Write a program for "Connectionless Iterative Service" in which the server finds the factorial of a number cent by the client and sends it back.
- AIM: TO IMPLEMENT "CONNECTIONLESS ITERATIVE SERVICE" TO FIND THE FACTORIAL OF A NUMBER SENT BY THE CLIENT USING UDP IN CPP.
- THEORY: 1. UDP CLIENT SERVER: In UDP, the client does not form a connection with the server (hence, connectionless) like in TCP. Instead, the client just sends a datagram.

 Similarly, the server boes not need to accept a connection and just waits for datagrams to arrive. Datagrams upon arrival contain the address of sender which the server uses to send data to the correct client.
 - 2. CONNECTIONLESS ITERATIVE SERVER: In this model, the server receives 2 request packet from UDP, processes the request 2nd gives response to the UDP to send to the client. The packets are stored in queue and processed in order of arrival.
 - 3. CONNECTIONLESS SERVICE: It is a technique that is used in data communications to send or transfer data or message at layer 4, i.e., Transport layer of Open System Interconnection model.

4. ITERATIVE SERVER: An iterative server processes request from clients in a serial manner; one connection is served and responded to before the server accepts a new client connection.

The entire operation can be broken down as follows:

CONNECTIONLESS ITERATIVE SERVICE SERVER:

- (i) A socket is created and binded to an advertised
- (ii) An infinite loop is started to process the client requests for connections.
- AND ARE PERFERENCE TOWNS PARME AND SOUTH ARE SOUTHER THE WORK MANAGE
- (iii) The process receives a number from the client using recvfronce) function and calculates the factorial for the received number, and cends it back to the client using sendto() function

CONNECTIONLESS ITERATIVE SERVICE CLIENT:

- (i) A socket is created and binded.
- (ii) The number, whose factorial is to be found, is sent as a message from the user (client) using sendto() function.
- (iii) The factorial value from the server is received using recufrom () function and displayed.

CODE : 11 CIS SERVER # include <iostresm> # include < estallib.> # include < cstring > # include < unista. h> # include < systtypes.h> # include < sys | socket. h> # include < 2rp2 | inet.h> # include < netinet lin.h> # define PORT 8080 # define MAXLINE LOZL Char fact [MAXLINE]; using namespace std; int multiply (int xx, int res[], int ressize) { int carry = 0; for (int ii = 0; ii < ressize; ++ii) {

int prod = res [ii] * xx + czrry;

res ["] = prod 1. 10;

czrry = prod 1 10;

```
while (carry) {
            res [ressize] = carry 1.10;
            Carry = carry 110;
     15 shows and ++ yes Size; . while to said America
      return ressize;
               void factorial (int number) {
Mint res [MAXLINE]; 1 30 30 10
( char num To Tat [5];
  ( res [o] = 1 ( )
         int ressize = I;
         for (int nx = 2; nx <= number; ++ xx)
        ressize = multiply (nx, res, ressize);
         for ( int " = ressize - 1; ">= 0; -- ") {
          sprintf (numToTat, "1.d", res [ii]);
               streat (fact, numToTat);
                        VI/ LAVE BOMONIMO
         }
f
int main () {
     int sockfd;
     struct sockaddr-in servaddr, cliaddr:
           STOREST STATE THE STATE
     if ((sockfd = socket (AF-INET, SOCK-DGRAM, O)) <0) {
            perror ("FAIL: SOCKET CREATION In");
            exit (EXIT-FATIURE);
```

```
cout « " SUCCESS: SOCKET CREATED IN";
memset (& servaddr, O, sizeof (servaddr));
memset (& cliador, O, sizeof (cliador));
 servaddr. sin-family = AF_INET;
 servaddr. sin-2ddr. s-2ddr = INADDR-ANY;
 servaddr. sin-port = htons (PORT);
 if (bind (sockfd, (const struct sockaddr *) & servaddr,
                          size of (servaddr)) < 0)
           perror ("FAIL: SERVER BIND In");
            exit (EXIT-FAILURE);
  Z
 cout « "SUCCESS: SERVER BOUNDIN";
 cout «"SERVER LIGTENING FOR MESSAGES ... IN'M";
 Char buffer EMAXLINE ];
  unsigned int len, nn;
  Len = size of (clizddr);
  while (1) {
         memset (buffer, O, MAXLINE);
         bzero (fact, MAXLINE);
         nn = recufron (socktd, (char *) buffer,
               MAXITHE, MSG-WAITALL, (Struct socksiddy*)
                & clisadar, & len);
         buffer [nn] = '10';
        Cout < " REQUEST RECEIVED: FACTORIAL OF " « buffer
         int num = 2 to: (buffer);
```

```
factorial (num);
    bzero (buffer, MAXLINE);
      stropy (buffer, fact);
      sendto (sockfd, (const char *) buffer, strlen (buffer),
            MSG-CONFIRM, (const struct sockadd+ *) bcliaddr,
       (10/ (2 len); (32)
  close (sockfd);
11 CIS CLIENTA
# include <iostream>
# include <<stdlib>
# include <unista.h>
# include 4 estring>
# include < systtypes.h>
# include < sys | socket. h >
# include < arpalinet.h>
# include < netinet lin. h >
# define PORT 8080
# define MAXLINE 1024
using namespace std;
int main () {
     int sockfd;
```

```
Struct sockaddr_in servaddr;
if ((sockfd = socket (AF-INET, SOCK-DGRAM, O)) < 0) {
        perror ("FAIL: SOCKET CREATION In");
        exit (EXIT-FATIURE);
1
Cout << "SUCCESS: SOCKET CREATED ININ";
memset (& servaddr, O, sizeof (servaddt));
 servaddr. sin-family = AF-INET;
 servaddr. sin - port = htons (PORT);
 Servaddr. Sin-2ddr. S-2ddr = INADDR-ANY;
 unsigned int nn, len;
 Char buffer [MAXLINE], mag [MAXLINE];
 COLL << " ENTER NUMBER WHOSE FACTORTAL IS TO BE FOUND ";
 while (1) }
      menset (msg, O, MAXLINE);
      memset (buffer, O, MAXINE);
      cout << "\n> ";
      fgets (msg, MAXLINE, stdin);
     sendto (sockfd, (const char *) msg, strlen (msg), Msg-Confirm
             (const struct sockaddy *) & servaddr, sizeof (servaddr));
     nn = recuform (sockfd, (char *) buffer, MAXLINE, MSG-WAITALL
           (struct sockaddr *) & servaddr, &len);
     buffer [nn] = 101;
     if (strnemp (msg, "exit", 4) == 0)
              brezk;
     cout << "FACTORIAL IS: " < buffer;
 close (sockfd);
  return 0;
```

// CONNECTIONLESS ITERATIVE SERVICE SERVER

```
**subhojit1912160@GrimBook-Orcen-15:/mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP*
g++ cisServer.cpp -o server
subhojit1912160@GrimBook-Orcen-15:/mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP$
g++ cisServer.cpp -o server
subhojit1912160@GrimBook-Orcen-15:/mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP$
J/server
SUCCESS: SOCKET CREATED
SUCCESS: SOCKET CREATED
SUCCESS: SERVER BOUND
SERVER LISTENING FOR MESSAGES TO ECHO BACK...

REQUEST RECEIVED: FACTORIAL OF 4
REQUEST RECEIVED: FACTORIAL OF 6
REQUEST RECEIVED: FACTORIAL OF 10
REQUEST RECEIVED: FACTORIAL OF 10
REQUEST RECEIVED: FACTORIAL OF 50
REQUEST RECEIVED: FACTORIAL OF 50
REQUEST RECEIVED: FACTORIAL OF 4
REQUEST RECEIVED: FACTORIAL OF 6
REQUEST RECEIVED: F
```

// CONNECTIONLESS ITERATIVE SERVICE CLIENT

```
🙏 subhojit1912160@GrimBook-Orcen-15: /mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP
                       ok-Orcen-15:/mnt/d/Documents/NITS/5th Sem/Online Classes/LAB C5311 Computer Networks/LAB 4/CPP$
g++ cisClient.cpp -o client
      it1912160@GrimBook-Orcen-15./mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP$
 ./client
SUCCESS: SOCKET CREATED
ENTER NUMBER WHOSE FACTORIAL IS TO BE FOUND
FACTORIAL IS : 24
FACTORIAL IS : 6
FACTORIAL IS: 720
FACTORIAL IS : 3628800
FACTORIAL IS : 933262154439441526816992388562667004907159682643816214685929638952175999932299156089414639761565182862
53697920827223758251185210916864000000000000000000000000
FACTORIAL IS: 30414093201713378043612608166064768844377641568960512000000000000
FACTORIAL IS : 120
FACTORIAL IS: 24
FACTORIAL IS : 1
> exit
subhojit1912160@GrimBook-Orcen-15:/mnt/d/Documents/NITS/5th Sem/Online Classes/LAB CS311 Computer Networks/LAB 4/CPP$
```

Output Explanation:

Firstly, the CIS Server is compiled and run, making it open to receive all types of connections. Then the CIS Client is compiled and run, connecting it to the CIS Server. Then, in the Client Terminal, the number for which the factorial is to be found is entered. The Server Terminal receives the request and calculates the factorial for the requested number. After the factorial is calculated, the server echoes back the calculated value (which is the factorial value) to the client, which is then displayed on the Client Terminal.