PROGRAM-4

Implement client-server communication using socket programming and UDP as transport layer protocol

AIM: To implement client server program using UDP as transport layer protocol

ALGORITHM

UDP Server :

1. Create a UDP socket.
2. Bind the socket to the server address.
3. Wait until the datagram packet arrives from the client.
4. Process the datagram packet and send a reply to the client.

UDP Client :

1. Create a UDP socket.
2. Send a message to the server.
3. Wait until response from the server is received.

CLIENT PROGRAM

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>

int main(){
  int clientSocket, portNum, nBytes;
  char buffer[1024];
```

```
struct sockaddr_in serverAddr;
socklen_t addr_size;
/*Create UDP socket*/
clientSocket = socket(AF_INET, SOCK_DGRAM, 0);

/*Configure settings in address struct*/
serverAddr.sin_family = AF_INET;
serverAddr.sin_port = htons(7891);
serverAddr.sin_addr.s_addr = INADDR_ANY;

/*Initialize size variable to be used later on*/
addr_size = sizeof(serverAddr);

while(1){
  printf("Type a sentence to send to server:\n");
  fgets(buffer,1024,stdin);
  printf("You typed: %s",buffer);
  nBytes = strlen(buffer) + 1;
     /*Send message to server*/
  sendto(clientSocket,buffer,nBytes,0,(struct sockaddr *)&serverAddr,addr_size);

  /*Receive message from server*/
          nBytes = recvfrom(clientSocket,buffer,1024,0,NULL, NULL);
  printf("Received from server: %s\n",buffer);
}
```

```
  return 0;
}
```

SERVER PROGRAM

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
#include <stdlib.h>

int main(){
  int udpSocket, nBytes;
  char buffer[1024];
  struct sockaddr_in serverAddr, clientAddr;
  //struct sockaddr_storage serverStorage;
  socklen_t addr_size, client_addr_size;
  int i;

  /*Create UDP socket*/
  udpSocket = socket(AF_INET, SOCK_DGRAM, 0);

  /*Configure settings in address struct*/
  serverAddr.sin_family = AF_INET;
  serverAddr.sin_port = htons(7891);
   serverAddr.sin_addr.s_addr = INADDR_ANY;
```

```
  /*Bind socket with address struct*/
  bind(udpSocket, (struct sockaddr *) &serverAddr, sizeof(serverAddr));

  /*Initialize size variable to be used later on*/
  addr_size = sizeof(serverAddr);

  while(1){
    /* Try to receive any incoming UDP datagram. */
    nBytes = recvfrom(udpSocket,buffer,1024,0,(struct sockaddr *)&serverAddr,
&addr_size);
    printf("%s",buffer);

    printf("enter the message to client");
    fgets(buffer,1024,stdin);

    sendto(udpSocket,buffer,nBytes,0,(struct sockaddr *)&serverAddr,addr_size);
  }

  return 0;
}
```

RESULT: Client server program using UDP as transport layer protocol is implemented and output is obtained.