

RA1811031010010
Swapnanil Dhol
Computer Networks Lab

Server Code:

```
#include<stdio.h>
#include<stdlib.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<unistd.h>
#include<string.h>
#define MAXLINE 1024
#define PORT 5035

int main(){

    int socketDescriptor = socket(AF_INET, SOCK_DGRAM, 0);
    int number;
    socklen_t addressLength;
    char recvBuffer[MAXLINE],sendBuffer[MAXLINE];
    pid_t cpid;

    struct sockaddr_in serverAddress,clientAddress;
    serverAddress.sin_family = AF_INET;
    serverAddress.sin_addr.s_addr=INADDR_ANY;
    serverAddress.sin_port=htons(PORT);

    bind(socketDescriptor,(struct sockaddr*)&serverAddress,sizeof(serverAddress));

    printf("\nServer Started ...\n");

    cpid=fork();

    if(cpid==0)
    {

        while(1){

            bzero(&recvBuffer,sizeof(recvBuffer));

            addressLength = sizeof(clientAddress);

            recvfrom(socketDescriptor,recvBuffer,sizeof(recvBuffer),0,(struct
sockaddr*)&clientAddress,&addressLength);

            printf("\n Client's Message: %s ",recvBuffer);
        }}
    else
    {
        while(1)
```

```

{
    //if(number<6)
    //perror("send error");
    addressLength = sizeof(serverAddress);
    printf("\nType a message here ... ");
    fgets(sendBuffer,10000,stdin);
    sendto(socketDescriptor,sendBuffer,strlen(sendBuffer)+1,0,(struct
sockaddr*)&clientAddress,addressLength);
    }}return 0;}

```

Client Code:

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include<netdb.h>
#include<string.h>

```

```

#define MAXLINE 1024
#define PORT 5035

```

```

int main(){

```

```

    int socketDescriptor = socket(AF_INET, SOCK_DGRAM, 0);
    socklen_t addressLength;
    pid_t cpid;

```

```

    char sendBuffer[MAXLINE],recvBuffer[MAXLINE];

```

```

    struct sockaddr_in serverAddress;
    serverAddress.sin_family = AF_INET;
    serverAddress.sin_addr.s_addr = INADDR_ANY;
    serverAddress.sin_port = htons(PORT);

```

```

    addressLength = sizeof(serverAddress);

```

```

    connect(socketDescriptor,(struct sockaddr*)&serverAddress,addressLength);

```

```

    cpid=fork();
    if(cpid==0)
    {
        while(1)
        {
            bzero(&sendBuffer,sizeof(sendBuffer));
            addressLength = sizeof(serverAddress);

```

```

printf("\nType a message here ... ");
fgets(sendBuffer,10000,stdin);
sendto(socketDescriptor,sendBuffer,strlen(sendBuffer)+1,0,(struct
sockaddr*)&serverAddress,addressLength);
}
}
else{
while(1)
{
bzero(&recvBuffer,sizeof(recvBuffer));
addressLength = sizeof(serverAddress);
recvfrom(socketDescriptor,recvBuffer,sizeof(recvBuffer),0,(struct
sockaddr*)&serverAddress,&addressLength);

printf("\nServer's Message: %s\n",recvBuffer);
}
}
return 0;
}

```

