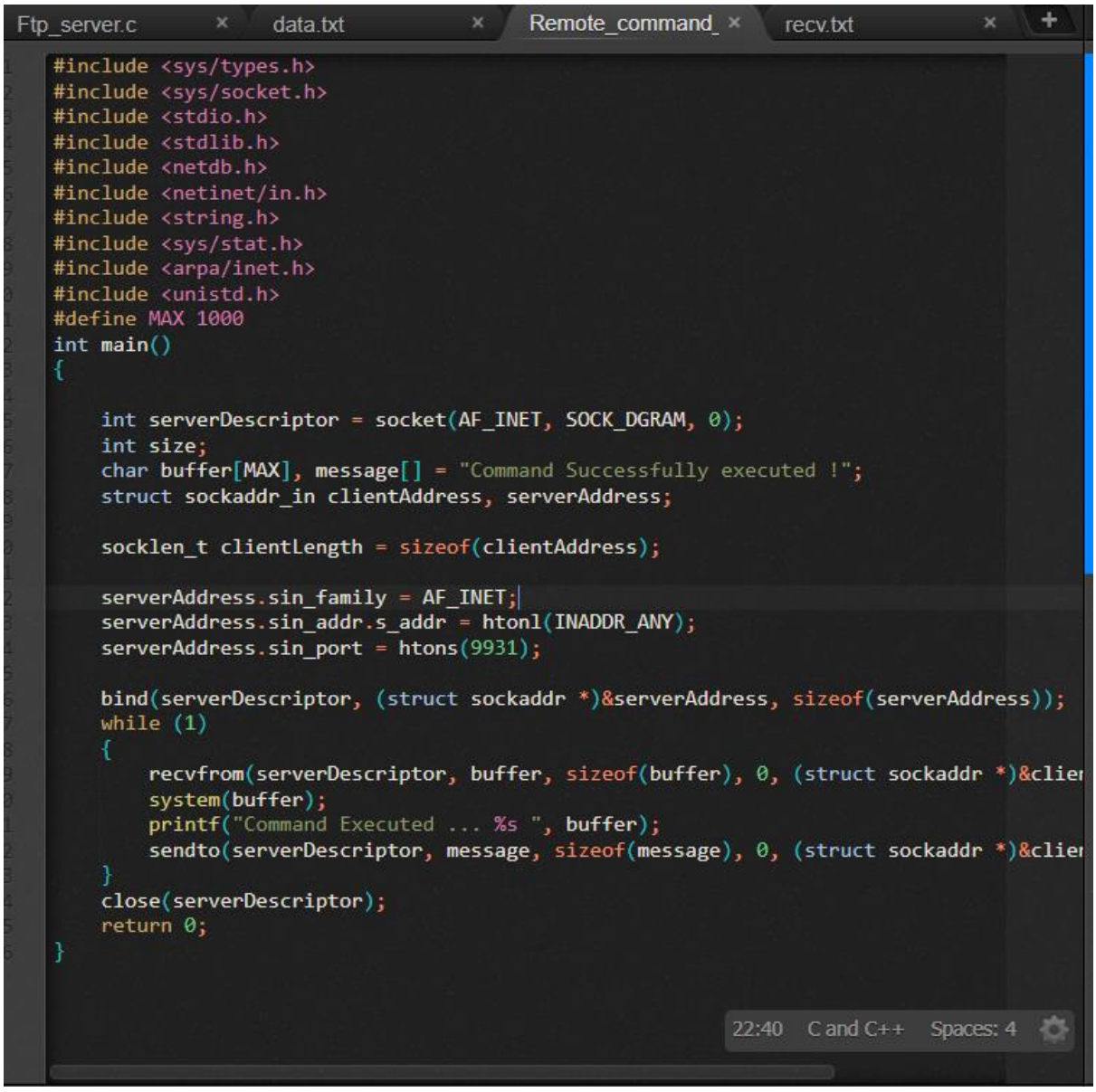
18CSC302J (Computer Networks Lab)
Lab session - Remote Command Execution Using UDP

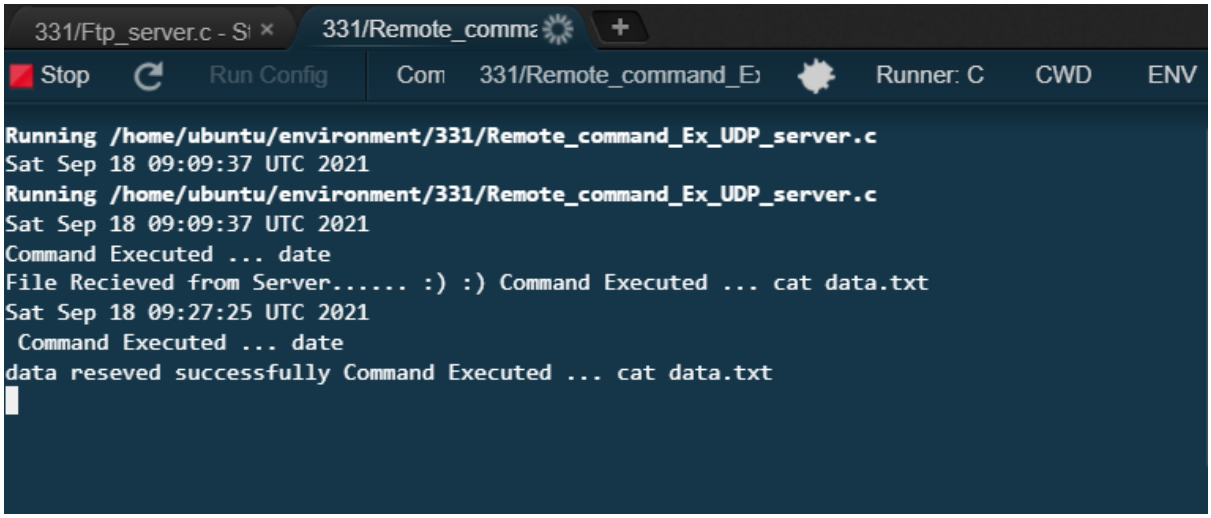
Name :- Puneet Sharma
Reg. No. :- RA1911003010331
Class :-CSE F1

SERVER CODE:

```
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <stdlib.h>
#include <netdb.h>
#include <netinet/in.h>
#include <string.h>
#include <sys/stat.h>
#include <arpa/inet.h>
#include <unistd.h>
#define MAX 1000
int main()
{
   int serverDescriptor = socket(AF_INET, SOCK_DGRAM, 0);
   int size;
   char buffer[MAX], message[] = "Command Successfully executed !";
   struct sockaddr_in clientAddress, serverAddress;
   socklen_t clientLength = sizeof(clientAddress);
   serverAddress.sin_family = AF_INET;
   serverAddress.sin_addr.s_addr = htonl(INADDR_ANY);
   serverAddress.sin_port = htons(9931);
   bind(serverDescriptor, (struct sockaddr *)&serverAddress, sizeof(serverAddress));
   while (1)
   {
```

```
    recvfrom(serverDescriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&clientAddress,
&clientLength);

    system(buffer);

    printf("Command Executed ... %s ", buffer);

    sendto(serverDescriptor, message, sizeof(message), 0, (struct sockaddr *)&clientAddress,
clientLength);

  }

  close(serverDescriptor);

  return 0;

}
```





CLIENT CODE

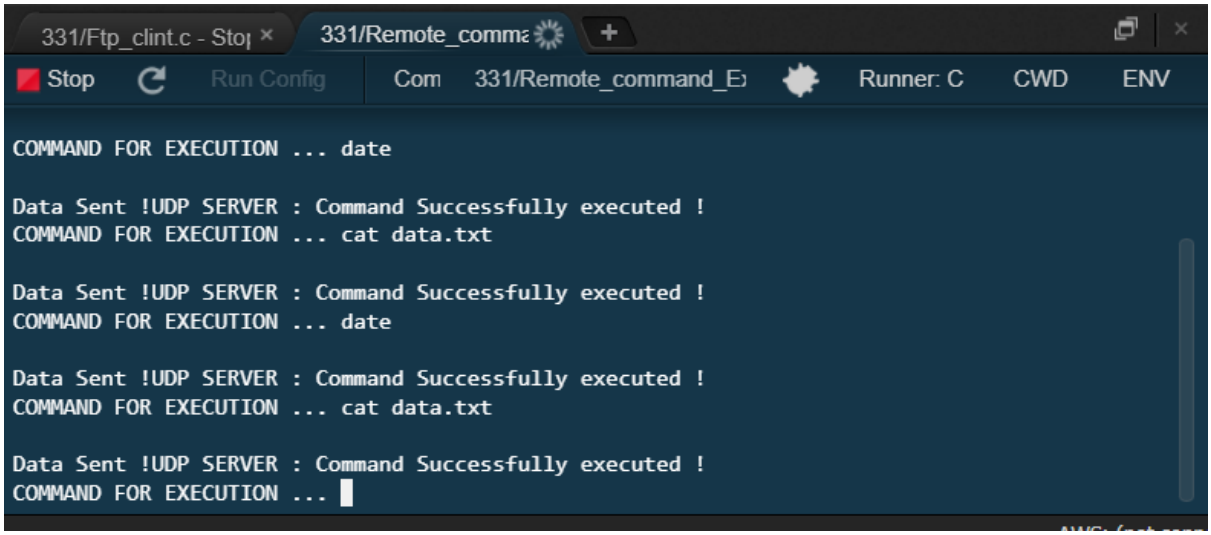
```
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <unistd.h>
#include <netdb.h>
#include <netinet/in.h>
#include <string.h>
#include <arpa/inet.h>
#define MAX 1000
int main()
{
   int serverDescriptor = socket(AF_INET, SOCK_DGRAM, 0);
   char buffer[MAX], message[MAX];
   struct sockaddr_in cliaddr, serverAddress;
   socklen_t serverLength = sizeof(serverAddress);
   bzero(&serverAddress, sizeof(serverAddress));
   serverAddress.sin_family = AF_INET;
   serverAddress.sin_addr.s_addr = inet_addr("127.0.0.1");
   serverAddress.sin_port = htons(9931);
   bind(serverDescriptor, (struct sockaddr *)&serverAddress, sizeof(serverAddress));
   while (1)
   {
```

```
        printf("\nCOMMAND FOR EXECUTION ... ");
        fgets(buffer, sizeof(buffer), stdin);
        sendto(serverDescriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&serverAddress,
serverLength);
        printf("\nData Sent !");
        recvfrom(serverDescriptor, message, sizeof(message), 0, (struct sockaddr *)&serverAddress,
&serverLength);
        printf("UDP SERVER : %s", message);
    }
    return 0;
}
```

Ftp_clint.c × Remote_command_ ×

```c
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <unistd.h>
#include <netdb.h>
#include <netinet/in.h>
#include <string.h>
#include <arpa/inet.h>
#define MAX 1000

int main()
{
    int serverDescriptor = socket(AF_INET, SOCK_DGRAM, 0);
    char buffer[MAX], message[MAX];
    struct sockaddr_in cliaddr, serverAddress;
    socklen_t serverLength = sizeof(serverAddress);

    bzero(&serverAddress, sizeof(serverAddress));
    serverAddress.sin_family = AF_INET;
    serverAddress.sin_addr.s_addr = inet_addr("127.0.0.1");
    serverAddress.sin_port = htons(9931);

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

    while (1)
    {
        printf("\nCOMMAND FOR EXECUTION ... ");
        fgets(buffer, sizeof(buffer), stdin);
        sendto(serverDescriptor, buffer, sizeof(buffer), 0, (struct sockaddr *)&serverA
        printf("\nData Sent !");
        recvfrom(serverDescriptor, message, sizeof(message), 0, (struct sockaddr *)&ser
        printf("UDP SERVER : %s", message);
    }
    return 0;
}
```

35:2 C and C++ Spaces: 4


331/Ftp_clint.c - Stop
331/Remote_comma
Stop
Run Config
Com
331/Remote_command_E
Runner: C
CWD
ENV
COMMAND FOR EXECUTION ... date
Data Sent !UDP SERVER : Command Successfully executed !
COMMAND FOR EXECUTION ... cat data.txt
Data Sent !UDP SERVER : Command Successfully executed !
COMMAND FOR EXECUTION ... date
Data Sent !UDP SERVER : Command Successfully executed !
COMMAND FOR EXECUTION ... cat data.txt
Data Sent !UDP SERVER : Command Successfully executed !
COMMAND FOR EXECUTION ...