Experiment 4a

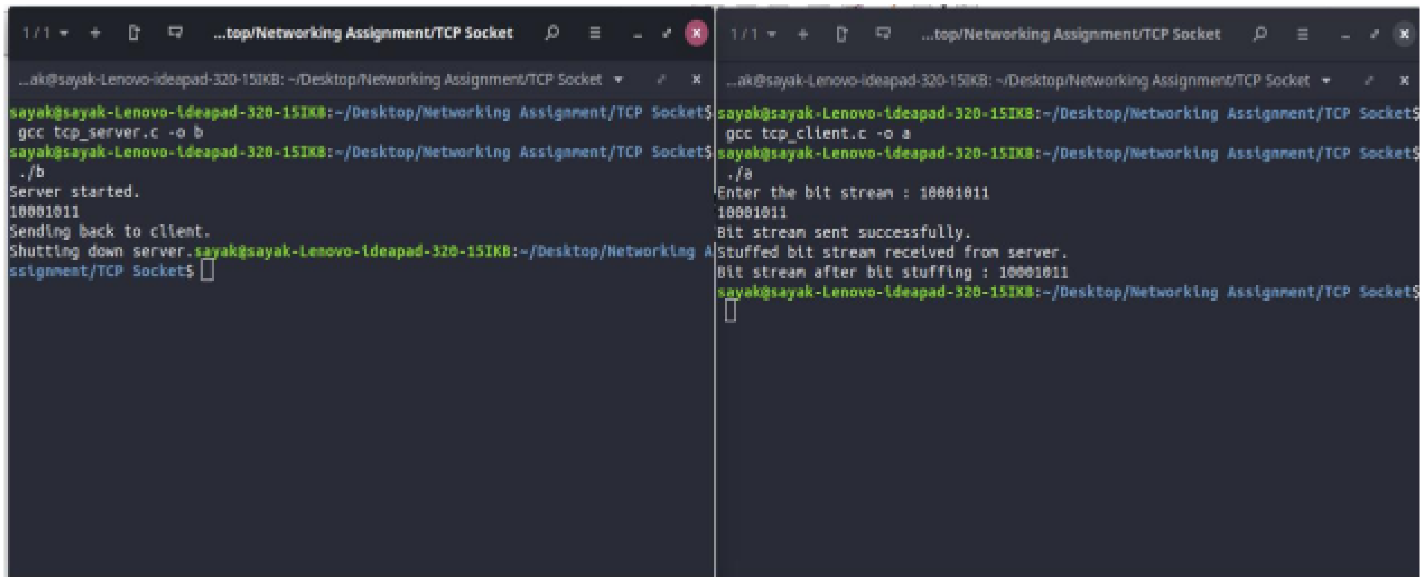
client.c

```
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>
int main(){
    int sockfd;
    int len;
    struct sockaddr_in address;
    int result;
    char ch = 'A';
    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    address.sin_family = AF_INET;
    address.sin_addr.s_addr = inet_addr("127.0.0.1");
    address.sin_port = 9734;
    len = sizeof(address);
    result = connect(sockfd, (struct sockaddr *)&address, len);
    if(result == -1) {
        perror("oops: client1");
        exit(1);}
    else{
        char *bit = (char *)calloc(16, sizeof(char));
        printf("Enter the bit stream : ");
        fflush(stdin);
        scanf("%[^\n]%*c",bit);
        bit[15]='\0';
        int i=0,c=0;
        while(bit[i]!='\0'){
        if(bit[i]=='1' || bit[i]=='0'){i++; continue;}
        else{printf("Invalid bit stream."); c=1; i++;}
        if(c==0){
            printf("%s\n",bit);
            write(sockfd, bit, strlen(bit));
            printf("Bit stream sent successfully.\n");
            char *output=(char *)calloc(20,sizeof(char));
            read(sockfd, output, 20);
        printf("Bit stream received after bit stuffing: ");
            printf("%s\n",output);}}}
```

server.c

```
#include <sys/types.h>
#include <sys/socket.h>
#include <stdio.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>
int main(){
     int cid, server_sockfd;
     int server_len, client_len;
     struct sockaddr_in server_address,s,c;
     struct sockaddr_in client_address;
     server_sockfd = socket(AF_INET, SOCK_STREAM, 0);
     server_address.sin_family = AF_INET;
     server_address.sin_addr.s_addr = inet_addr("127.0.0.1");
     server_address.sin_port = 9734;
     server_len = sizeof(server_address);
bind(server_sockfd, (struct sockaddr *)&server_address,
server_len);
     listen(server_sockfd, 5);
     printf("Server started.\n");
     int l=sizeof(c);
     cid = accept(server_sockfd, (struct sockaddr*)&s, &l);
     char *bit_stream = (char *)calloc(16, sizeof(char));
     char *out= (char *)calloc(20, sizeof(char));
     read(cid, bit_stream, 16);
     int i=0,j=0, count=0,loc=-99;
     while(bit_stream[i]!='\0'){
          out[j]=bit_stream[i];
          if(bit_stream[i]=='0'){ count=0;}
          if(bit_stream[i]=='1'){
               count++;
               if(count==5){loc=i;}}
          if(loc!=-99){
               out[loc+1]='0';
               j=loc+1;
               count=0;
               loc=-99;}j++; i++;}
     printf("%s\n",out);
     printf("Sending back to client.\n");
     write(cid,out,j);
     printf("Shutting down server.\n");}
```

OUTPUT:


1 / 1
...top/Networking Assignment/TCP Socket
...ak@sayak-Lenovo-ideapad-320-15IKB: ~/Desktop/Networking Assignment/TCP Socket
1 / 1
...top/Networking Assignment/TCP Socket
...ak@sayak-Lenovo-ideapad-320-15IKB: ~/Desktop/Networking Assignment/TCP Socket

```
sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking Assignment/TCP Socket$
 gcc tcp_server.c -o b
sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking Assignment/TCP Socket$
 ./b
Server started.
10001011
Sending back to client.
Shutting down server.sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking A
ssignment/TCP Socket$ 
```

```
sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking Assignment/TCP Socket$
 gcc tcp_client.c -o a
sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking Assignment/TCP Socket$
 ./a
Enter the bit stream : 10001011
10001011
Bit stream sent successfully.
Stuffed bit stream received from server.
Bit stream after bit stuffing : 10001011
sayak@sayak-Lenovo-ideapad-320-15IKB:~/Desktop/Networking Assignment/TCP Socket$
 
```