

CSE 1004
LAB 4
FLOW CONTROL - ARQ

Name: Ayush Sharma

Reg. No: 15BCE1335

Code:

Receiver Side

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <string.h>
int main(int argc, char *argv[])
{
    char buffer[6];
    buffer[5]='\0';
    int sockfd,a,connfd,len;
    struct sockaddr_in servaddr,cliaddr;
    const char akg='p';
    sockfd=socket(AF_INET,SOCK_STREAM,0);
    if(sockfd==-1)
        printf("Error creating socket\n");
    bzero(&servaddr,sizeof(servaddr));
    servaddr.sin_port=htons(PORT);
    servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
    bind(sockfd,(struct sockaddr*)&servaddr,sizeof(servaddr));
    if((a=listen(sockfd,5))<0)
        printf("Error in LISTEN function")
        while(TRUE)
        {
            connfd=accept(sockfd,(struct sockaddr*)&cliaddr,&len);
            printf("Accepted client request");
            while(TRUE){
                strcpy(buffer,"");
                read(connfd,buffer,4);
                printf("\nMsg received :%s",buffer);
                int delaysms=rand() % 4;
                printf("\nSimulating delay of %d s before sending AKG.",delaysms);
                sleep(delaysms);
                write(connfd,&akg,1);
                printf("\nSent AKG\n\n");
            }
        }
    close(sockfd);
}
```

Server Side

```
#include <stdio.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <arpa/inet.h>
typedef enum {
    SEND,
    WAIT,
    OVER
} state;
int main(int argc, char *argv[])
{
    int sockfd;
    struct sockaddr_in serv_addr;
    char buff[20];
    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    memset(&serv_addr, '0', sizeof(serv_addr));
    serv_addr.sin_family = AF_INET;
    serv_addr.sin_port = htons(PORT);
    serv_addr.sin_addr.s_addr = inet_addr("127.0.0.1");
    const struct timeval sock_timeout = { .tv_sec = 2, .tv_usec = 0 };
    setsockopt(sockfd, SOL_SOCKET, SO_RCVTIMEO, (char*)&sock_timeout, sizeof(sock_timeout));
    if (connect(sockfd, (struct sockaddr*)&serv_addr, sizeof(serv_addr)) < 0)
    {
        printf("\n Error : Connect Failed \n");
        return 1;
    }
    state curstate = SEND;
    printf("Timeout is 2 secs\n");
    printf("Enter the string to send\n");
    fgets(buff, sizeof(buff), stdin);
    int datalen = strlen(buff);
    int datasent = 0;
    char akg;
    //ARQ GOES HERE
    while(1){
        switch (curstate) {
            case SEND:
                printf("Sending Frame %d\n", datasent/4+1);
                write(sockfd, &buff[datasent], 4);
                curstate = WAIT;
                break;
            case WAIT:
                recv(sockfd, &akg, 1, 0);
                if(akg == 'p') //Positive reply
                {
```

```

    akg=' ':
    datasent+=4;
    if(datasent>=datalen){
    printf("Successful Transmission\n");
    return 0;
    }
    else
    curstate=SEND;
}
else {
    printf("Timeout reached\n");
    curstate=SEND;
}
break;
};
}

close(sockfd);

return 0;
}

```

The screenshot shows two terminal windows side-by-side, both running on a system with the username 'guest-73txru' and the directory '~/Downloads'.

Left Terminal (Receiver):

```

guest-73txru@hp-lappy: ~/Downloads
guest-73txru@hp-lappy:~/Downloads$ ./receiver
Server started
Accepted client request
Msg received :Ayuse
Simulating delay of 3 s before sending ACK.
Sent ACK

Msg received :Ayuse
Simulating delay of 2 s before sending ACK.
Sent ACK

Msg received :h 13s
Simulating delay of 1 s before sending ACK.
Sent ACK

Msg received :35
Simulating delay of 3 s before sending ACK.
Sent ACK

Msg received :
guest-73txru@hp-lappy:~/Downloads$

```

Right Terminal (Sender):

```

guest-73txru@hp-lappy:~/Downloads$ ./sender
Timeout is 2 secs
Enter the string to send
Ayush 1335
Sending Frame 1
Timeout reached
Sending Frame 1
Sending Frame 2
Sending Frame 3
Successful Transmission
guest-73txru@hp-lappy:~/Downloads$

```