Vishwakarma Institute of Technology, Pune

**Name:** Aditya Udhav Suryawanshi

**Div:** TY CS D **Roll No.:** 49 **Batch:** 2

**PRN No.:** 12111453 **Assignment No.** 7 **Subject:** CN LAB

**Problem Statement:** Write a program to simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in peer to peer mode and demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode. Further extend it to real implementation of Flow Control over TCP protocol.

**Go Back N Code:**

**Server Side:**

*#include*<stdio.h>

*#include*<sys/types.h>

*#include* <winsock2.h>

*#include*<string.h>

*#include*<time.h>

*#include*<stdlib.h>

*#include*<ctype.h>

*#define* W 5

*char* a[10];

*char* b[10];

*void* alpha9(*int*);

*int* main()

{

*int* s,f,wl,c*=*1,x,i*=*0,j,n,p*=*0,e*=*0;

*struct* sockaddr\_in ser;

    s*=*socket(AF\_INET,SOCK\_STREAM,0);

    ser.sin\_family*=*AF\_INET;

    ser.sin\_port*=*6500;

    ser.sin\_addr.s\_addr*=*inet\_addr("127.0.0.1");

    connect(s,(*struct* sockaddr *\**) *&*ser, *sizeof*(ser));

    printf("\nTCP Connection Established.\n");

    printf("\nEnter the number of Frames: ");

    scanf("%d",*&*f);

    alpha9(f);

    send(s,a,*sizeof*(a),0);

    strcpy(b,"Time Out ");

*while*(1)

    {

*for*(i*=*0;i*<*W;i*++*)

        {

            alpha9(c);

            send(s,a,*sizeof*(a),0);

*if*(c*<=*f)

            {

                printf("\nFrame %d Sent",c);

                c*++*;

            }

        }

        i*=*0;

        wl*=*W;

*while*(i*<*W)

        {

            recv(s,a,*sizeof*(a),0);

            p*=*atoi(a);

*if*(strcmp(a,b)*==*0)

            {

                e*=*c*-*wl;

*if*(e*<*f)

        {

            printf("\nTime Out, Resent Frame %d onwards",e);

        }

*break*;

            }

*else*

            {

*if*(p*<=*f)

                {

                    printf("\nFrame %s Acknowledged",a);

                    wl*--*;

                }

*else*

                {

*break*;

                }

            }

*if*(p*>*f)

            {

*break*;

            }

            i*++*;

        }

*if*(wl*==*0 *&&* c*>*f)

        {

            send(s,b,*sizeof*(b),0);

*break*;

        }

*else*

        {

            c*=*c*-*wl;

            wl*=*W;

        }

    }

    close(s);

*return* 0;

}

*void* alpha9(*int* *z*)

{

*int* k,i*=*0,j,g;

    k*=*z;

*while*(k*>*0)

    {

        i*++*;

        k*=*k*/*10;

    }

    g*=*i;

    i*--*;

*while*(z*>*0)

    {

        k*=*z*%*10;

        a[i]*=*k*+*48;

        i*--*;

        z*=*z*/*10;

    }

    a[g]*=*'\0';

}

**Go Back N Code:**

**Client Side:**

*#include*<stdio.h>

*#include*<sys/types.h>

*#include* <winsock2.h>

*#include*<string.h>

*#include*<time.h>

*#include*<stdlib.h>

*#include*<ctype.h>

*#define* W 5

*#define* P1 50

*#define* P2 10

*char* a[10];

*char* b[10];

*void* alpha9(*int*);

*int* main()

{

*struct* sockaddr\_in ser,cli;

*int* s,n,sock,i,j,c*=*1,f;

*unsigned* *int* s1;

    s*=*socket(AF\_INET,SOCK\_STREAM,0);

    ser.sin\_family*=*AF\_INET;

    ser.sin\_port*=*6500;

    ser.sin\_addr.s\_addr*=*inet\_addr("127.0.0.1");

    bind(s,(*struct* sockaddr *\**) *&*ser, *sizeof*(ser));

    listen(s,1);

    n*=sizeof*(cli);

    sock*=*accept(s,(*struct* sockaddr *\**)*&*cli, *&*n);

    printf("\nTCP Connection Established.\n");

    s1*=*(*unsigned* *int*) time(NULL);

    srand(s1);

    strcpy(b,"Time Out ");

    recv(sock,a,*sizeof*(a),0);

    f*=*atoi(a);

*while*(1)

    {

*for*(i*=*0;i*<*W;i*++*)

        {

            recv(sock,a,*sizeof*(a),0);

*if*(strcmp(a,b)*==*0)

            {

*break*;

            }

        }

        i*=*0;

*while*(i*<*W)

        {

            j*=*rand()*%*P1;

*if*(j*<*P2)

            {

                send(sock,b,*sizeof*(b),0);

*break*;

            }

*else*

            {

                alpha9(c);

*if*(c*<=*f)

                {

                    printf("\nFrame %s Received ",a);

                    send(sock,a,*sizeof*(a),0);

                }

*else*

                {

*break*;

                }

                c*++*;

            }

*if*(c*>*f)

            {

*break*;

            }

            i*++*;

        }

    }

    close(sock);

    close(s);

*return* 0;

}

*void* alpha9(*int* *z*)

{

*int* k,i*=*0,j,g;

    k*=z*;

*while*(k*>*0)

    {

        i*++*;

        k*=*k*/*10;

    }

    g*=*i;

    i*--*;

*while*(*z>*0)

    {

        k*=z%*10;

        a[i]*=*k*+*48;

        i*--*;

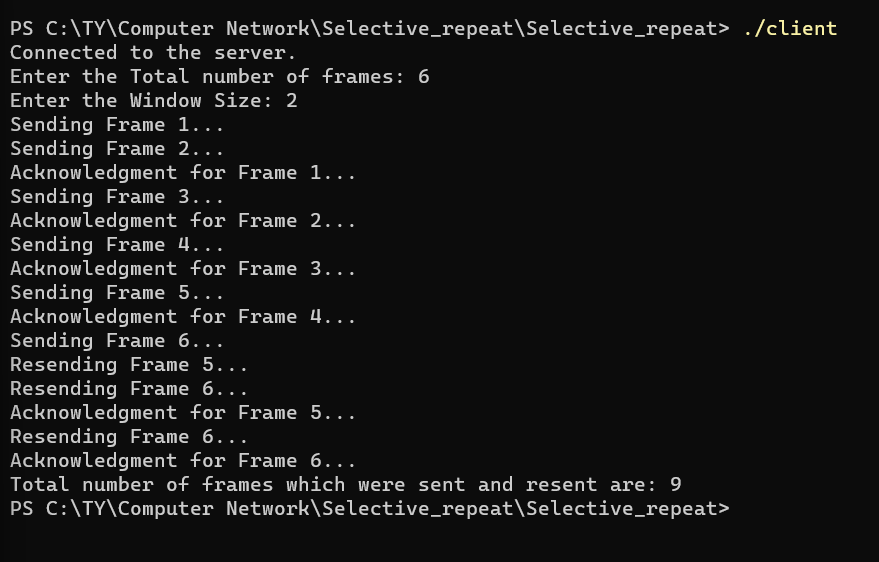
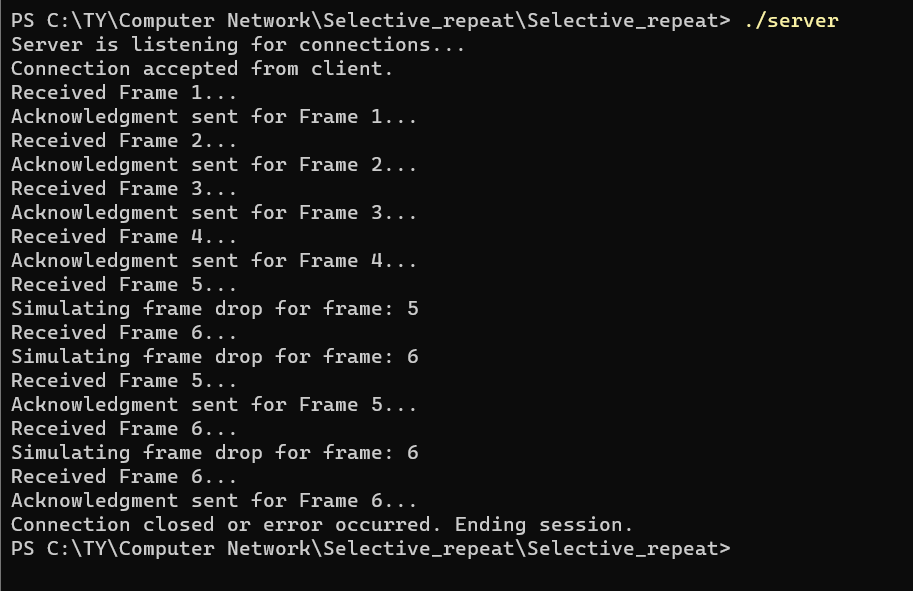
*z=z/*10;

    }

    a[g]*=*'\0';

}

**OUTPUT:**

****

**Selective Repeat Code:**

**Server Side:**

*#include*<stdio.h>

*#include*<sys/types.h>

*#include*<sys/socket.h>

*#include*<netinet/in.h>

*#include*<string.h>

*#include*<time.h>

*#include*<stdlib.h>

*#include*<ctype.h>

*#define* W 5

*char* a[10];

*char* b[10];

*void* alpha9(*int*);

*int* con();

*int* main()

{

*int* s,f,wl,c*=*1,x,i*=*0,j,n,p*=*0,e*=*0;

*struct* sockaddr\_in ser;

    s*=*socket(AF\_INET,SOCK\_STREAM,0);

    ser.sin\_family*=*AF\_INET;

    ser.sin\_port*=*6500;

    ser.sin\_addr.s\_addr*=*inet\_addr("127.0.0.1");

    connect(s,(*struct* sockaddr *\**) *&*ser, *sizeof*(ser));

    printf("\nTCP Connection Established.\n");

    printf("\nEnter the number of Frames: ");

    scanf("%d",*&*f);

    alpha9(f);

    send(s,a,*sizeof*(a),0);

    strcpy(b,"Time Out ");

*while*(1)

    {

*for*(i*=*0;i*<*W;i*++*)

        {

            alpha9(c);

            send(s,a,*sizeof*(a),0);

*if*(c*<=*f)

            {

                printf("\nFrame %d Sent",c);

                c*++*;

            }

        }

        i*=*0;

        wl*=*W;

*while*(i*<*W)

        {

            recv(s,a,*sizeof*(a),0);

            p*=*atoi(a);

*if*(a[0]*==*'N')

            {

                e*=*con();

*if*(e*<*f)

                {

                    printf("\nNAK %d",e);

                    printf("\nFrame %d sent",e);

                    i*--*;

                }

            }

*else*

            {

*if*(p*<=*f)

                {

                    printf("\nFrame %s Acknowledged",a);

                    wl*--*;

                }

*else*

                {

*break*;

                }

            }

*if*(p*>*f)

            {

*break*;

            }

            i*++*;

        }

*if*(wl*==*0 *&&* c*>*f)

        {

            send(s,b,*sizeof*(b),0);

*break*;

        }

*else*

        {

            c*=*c*-*wl;

            wl*=*W;

        }

    }

    close(s);

*return* 0;

}

*void* alpha9(*int* *z*)

{

*int* k,i*=*0,j,g;

    k*=z*;

*while*(k*>*0)

    {

        i*++*;

        k*=*k*/*10;

    }

    g*=*i;

    i*--*;

*while*(*z>*0)

    {

        k*=z%*10;

        a[i]*=*k*+*48;

        i*--*;

*z=z/*10;

    }

    a[g]*=*'\0';

}

*int* con()

{

*char* k[9];

*int* i*=*1;

*while*(a[i]*!=*'\0')

    {

        k[i*-*1]*=*a[i];

        i*++*;

    }

    k[i*-*1]*=*'\0';

    i*=*atoi(k);

*return* i;

}

**Selective Repeat Code:**

**Client Side:**

*#include*<stdio.h>

*#include*<sys/types.h>

*#include*<sys/socket.h>

*#include*<netinet/in.h>

*#include*<string.h>

*#include*<time.h>

*#include*<stdlib.h>

*#include*<ctype.h>

*#include*<arpa/inet.h>

*#define* W 5

*#define* P1 50

*#define* P2 10

*char* a[10];

*char* b[10];

*void* alpha9(*int*);

*void* alp(*int*);

*int* main()

{

*struct* sockaddr\_in ser,cli;

*int* s,n,sock,i,j,c*=*1,f;

*unsigned* *int* s1;

    s*=*socket(AF\_INET,SOCK\_STREAM,0);

    ser.sin\_family*=*AF\_INET;

    ser.sin\_port*=*6500;

    ser.sin\_addr.s\_addr*=*inet\_addr("127.0.0.1");

    bind(s,(*struct* sockaddr *\**) *&*ser, *sizeof*(ser));

    listen(s,1);

    n*=sizeof*(cli);

    sock*=*accept(s,(*struct* sockaddr *\**)*&*cli, *&*n);

    printf("\nTCP Connection Established.\n");

    s1*=*(*unsigned* *int*) time(NULL);

    srand(s1);

    strcpy(b,"Time Out ");

    recv(sock,a,*sizeof*(a),0);

    f*=*atoi(a);

*while*(1)

    {

*for*(i*=*0;i*<*W;i*++*)

        {

            recv(sock,a,*sizeof*(a),0);

*if*(strcmp(a,b)*==*0)

            {

*break*;

            }

        }

        i*=*0;

*while*(i*<*W)

        {

            L:

                j*=*rand()*%*P1;

*if*(j*<*P2)

            {

                alp(c);

                send(sock,b,*sizeof*(b),0);

*goto* L;

            }

*else*

            {

                alpha9(c);

*if*(c*<=*f)

                {

                    printf("\nFrame %s Received ",a);

                    send(sock,a,*sizeof*(a),0);

                }

*else*

                {

*break*;

                }

                c*++*;

            }

*if*(c*>*f)

            {

*break*;

            }

            i*++*;

        }

    }

    close(sock);

    close(s);

*return* 0;

}

*void* alpha9(*int* *z*)

{

*int* k,i*=*0,j,g;

    k*=z*;

*while*(k*>*0)

    {

        i*++*;

        k*=*k*/*10;

    }

    g*=*i;

    i*--*;

*while*(*z>*0)

    {

        k*=z%*10;

        a[i]*=*k*+*48;

        i*--*;

*z=z/*10;

    }

    a[g]*=*'\0';

}

*void* alp(*int* *z*)

{

*int* k,i*=*1,j,g;

    k*=z*;

    b[0]*=*'N';

*while*(k*>*0)

    {

        i*++*;

        k*=*k*/*10;

    }

    g*=*i;

    i*--*;

*while*(*z>*0)

    {

        k*=z%*10;

        b[i]*=*k*+*48;

        i*--*;

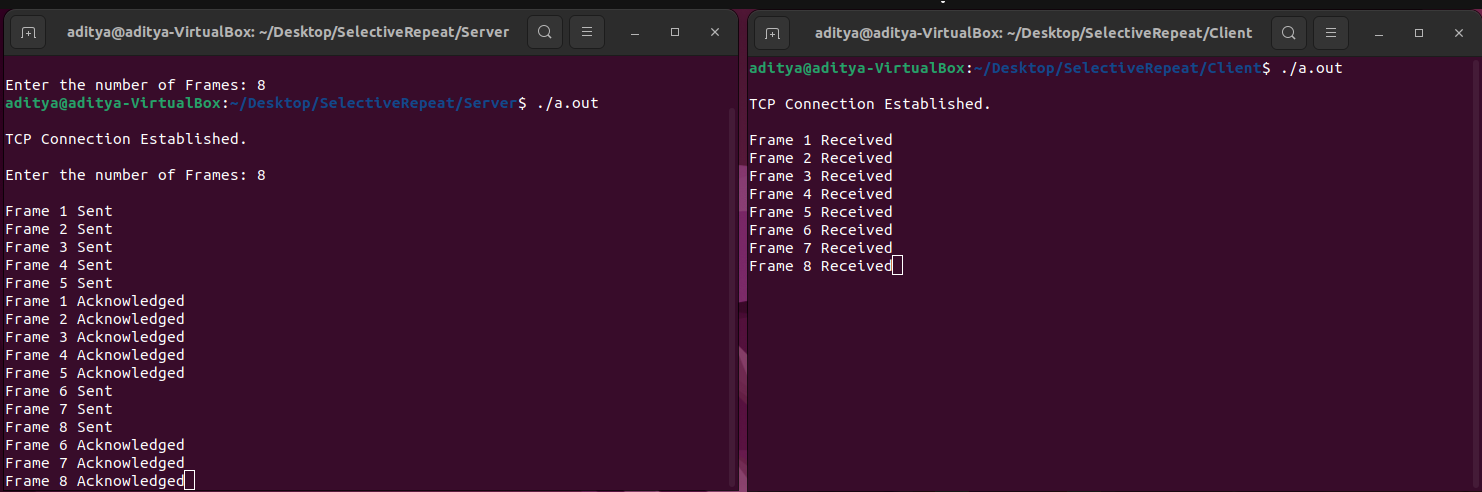
*z=z/*10;

    }

    b[g]*=*'\0';

}

**OUTPUT:**

****