

**Bansilal Ramnath Agarwal Charitable Trust’s**

**Vishwakarma Institute of Technology**

#### (An Autonomous Institute affiliated to Savitribai Phule Pune University)

**COMPUTER NETWORKS LAB ASSIGNMENTS**

|  |  |
| --- | --- |
| **NAME** | **ABBAS MADHVASWALA** |
| **PRN** | **12110285** |
| **DIVISION** | **CS-C** |
| **ROLL.NO** | **13** |
| **BATCH** | **B3** |

**LAB ASSIGNMENT NO : 03**

**PROBLEM STATEMENT**

Write a C/C++ 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.

**SOLUTION**

**Selective Repeat Modes:**

**Receiver Side Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <netinet/in.h>

#include <sys/socket.h>

#include <unistd.h>

struct transmit{

    int n;

    char c;

};

int main(){

    int server\_socket\_fd = socket(AF\_INET, SOCK\_STREAM, 0);

    if (server\_socket\_fd == -1){

        perror("Socket creation failed");

        return 1;

    }

    struct sockaddr\_in server\_addr;

    server\_addr.sin\_family = AF\_INET;

    server\_addr.sin\_port = htons(1245);

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

    if (bind(server\_socket\_fd, (struct sockaddr \*)&server\_addr, sizeof(server\_addr)) == -1){

        perror("Binding failed");

        close(server\_socket\_fd);

        return 1;

    }

    if (listen(server\_socket\_fd, 1) == -1){

        perror("Listening failed");

        close(server\_socket\_fd);

        return 1;

    }

    struct sockaddr\_in client\_addr;

    socklen\_t client\_len = sizeof(client\_addr);

    int socket\_fd = accept(server\_socket\_fd, (struct sockaddr \*)&client\_addr, &client\_len);

    if (socket\_fd == -1){

        perror("Accepting connection failed");

        close(server\_socket\_fd);

        return 1;

    }

    int maxpackets,n,i;

    printf("Enter the number of packets ");

    scanf("%d", &maxpackets);

    char packet [maxpackets];

    char data;

    i = 0;

    struct transmit t;

    while(i<maxpackets){

        if (recv(socket\_fd, &t, sizeof(t), 0) != -1) {

            n=t.n;data=t.c;

            if(n==i+1){

                send(socket\_fd, &n, sizeof(n), 0);

                printf("ack %d sent\n",n);

                packet[n-1]= data;

                i++;

            }

        }

    }

    for (int i = 0;i<maxpackets;i++){

        printf("\nThe %d packet's Data is %c ",i+1,packet[i]);

    }

    close(socket\_fd);

    close(server\_socket\_fd);

}

**Sender Side Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <netinet/in.h>

#include <sys/socket.h>

#include <unistd.h>

struct transmit{

    int n;

    char c;

};

int main(){

    int sockfd = socket(AF\_INET, SOCK\_STREAM, 0);

    if (sockfd == -1){

        perror("Socket creation failed");

        return 1;

    }

    struct sockaddr\_in server\_addr;

    server\_addr.sin\_family = AF\_INET;

    server\_addr.sin\_port = htons(1245);

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

    if (connect(sockfd, (struct sockaddr \*)&server\_addr, sizeof(server\_addr)) == -1){

        perror("Connection failed");

        close(sockfd);

        return 1;

    }

    int maxpackets,n;

    printf("Enter the number of packets ");

    scanf("%d", &maxpackets);

    int winsize;

    printf("Enter the number of packets in a window: ");

    scanf("%d", &winsize);

    char packet [maxpackets];

    char data;

    for (int i = 0;i<maxpackets;i++){

        printf("Enter the %d packet's Data: ",i+1);

        scanf(" %c", &data);

        packet[i] = data;

    }

    int ack = 0;

    int i = 0;

    struct transmit t;

    while(ack<maxpackets){

        // printf("%d %d %d",i,maxpackets,ack);

        if(i-ack<winsize&& i<maxpackets){

            i++;

            printf("\npacket %d send  with data %c",i,packet[i-1]);

            t.n=i;t.c=packet[i-1];

            send(sockfd, &t, sizeof(t), 0);

        }

        if (recv(sockfd, &n, sizeof(n), 0) != -1) {

            // printf("%d %d %d",i,n,ack);

            if(n==ack+1){

                ack++;

            }else{

                i = ack;

            }

            // printf("%d %d %d",i,n,ack);

        }

    }

    close(sockfd);

}

**Go and Back:**

**Sender Side Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <netinet/in.h>

#include <sys/socket.h>

#include <unistd.h>

int main(){

    int sockfd = socket(AF\_INET, SOCK\_STREAM, 0);

    if (sockfd == -1){

        perror("Socket creation failed");

        return 1;

    }

    struct sockaddr\_in server\_addr;

    server\_addr.sin\_family = AF\_INET;

    server\_addr.sin\_port = htons(1245);

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

    if (connect(sockfd, (struct sockaddr \*)&server\_addr, sizeof(server\_addr)) == -1){

        perror("Connection failed");

        close(sockfd);

        return 1;

    }

    int maxpackets,n;

    printf("Enter the number of packets ");

    scanf("%d", &maxpackets);

    int winsize;

    printf("Enter the number of packets in a window: ");

    scanf("%d", &winsize);

    char packet [maxpackets];

    char data;

    for (int i = 0;i<maxpackets;i++){

        printf("Enter the %d packet's Data: ",i+1);

        scanf(" %c", &data);

        packet[i] = data;

    }

    int ack = 0;

    int i = 0;

    while(ack<maxpackets){

        // printf("%d %d %d",i,maxpackets,ack);

        if(i-ack<winsize&& i<maxpackets){

            i++;

            printf("packet %d send  with data %c",i,packet[i-1]);

        }

        scanf("%d",&n);

        if (n!=-1){

            // printf("%d %d %d",i,n,ack);

            if(n==ack+1){

                ack++;

            }else{

                i = ack;

            }

            // printf("%d %d %d",i,n,ack);

        }

    }

}

**Receiver Side Code:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <netinet/in.h>

#include <sys/socket.h>

#include <unistd.h>

int main(){

    int server\_socket\_fd = socket(AF\_INET, SOCK\_STREAM, 0);

    if (server\_socket\_fd == -1){

        perror("Socket creation failed");

        return 1;

    }

    struct sockaddr\_in server\_addr;

    server\_addr.sin\_family = AF\_INET;

    server\_addr.sin\_port = htons(1245);

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

    if (bind(server\_socket\_fd, (struct sockaddr \*)&server\_addr, sizeof(server\_addr)) == -1){

        perror("Binding failed");

        close(server\_socket\_fd);

        return 1;

    }

    if (listen(server\_socket\_fd, 1) == -1){

        perror("Listening failed");

        close(server\_socket\_fd);

        return 1;

    }

    struct sockaddr\_in client\_addr;

    socklen\_t client\_len = sizeof(client\_addr);

    int socket\_fd = accept(server\_socket\_fd, (struct sockaddr \*)&client\_addr, &client\_len);

    if (socket\_fd == -1){

        perror("Accepting connection failed");

        close(server\_socket\_fd);

        return 1;

    }

    int maxpackets,n,i;

    printf("Enter the number of packets ");

    scanf("%d", &maxpackets);

    char packet [maxpackets];

    char data;

    i = 0;

    while(i<maxpackets){

        scanf("%d %c", &n,&data);

        if(n==i+1){

            printf("ack %d sent",n);

            packet[n-1]= data;

            i++;

        }

    }

    for (int i = 0;i<maxpackets;i++){

        printf("\nThe %d packet's Data is %c \n",i+1,packet[i]);

    }

}