Experiment 5: ARQ Mechanisms in DLL

Aim: To implement receiver algorithms for the different ARQ mechanisms at the Data Link Layer

Objective: After carrying out this experiment, students will be able to:

- implement receiver algorithms for the different ARQ mechanisms at the Data Link Layer
- Analyze the differences between the ARQ mechanisms

Problem statement: You are required to write a program that can receive frames at the data link layer. Assume that the user is entering the frames as the transmitter. You are required to implement stop and wait, go back N and selective repeat ARQ mechanisms. Consider that you have to transmit and receive a total of 20 frames using W_T=W_R=1, W_T=5 and W_R=1 and W_T=W_R=5 for stop and wait, go back N and selective repeat respectively

Analysis: While analyzing your program, you are required to address the following points:

- Difference between stop and wait, go back N and selective repeat.
- Comparison of the disadvantages of the different ARQ mechanisms.

MARKS DISTRIBUTION

Component	Maximum Marks	Marks Obtained
Preparation of Document	8	
Results	8	
Viva	4	
Total	**20**	

Submitted by: KAUSHAL VASHISTH

Register No: 18ETCS002147

© Ramaiah University of Applied Sciences

1. Algorithm/Flowchart

Receiver side algorithm for:

1. Stop and Wait:

Start

- Wait for frame arrival
- If arrived frame is the one expected, accept it and send an acknowledgement for it.
- If arrived frame is not of the sequence number that is expected, then wait for retransmission of correct frame.

Stop

2. Go back N

Start

- Wait for frame arrival.
- If several frames arrive in order, accept them all
- If a frame arrives out of order, send acknowledgement for the last correctly received frame. Discard all the subsequent frames arrived at the receiver.
- Expect frames to arrive starting at last incorrectly received frame.

Stop

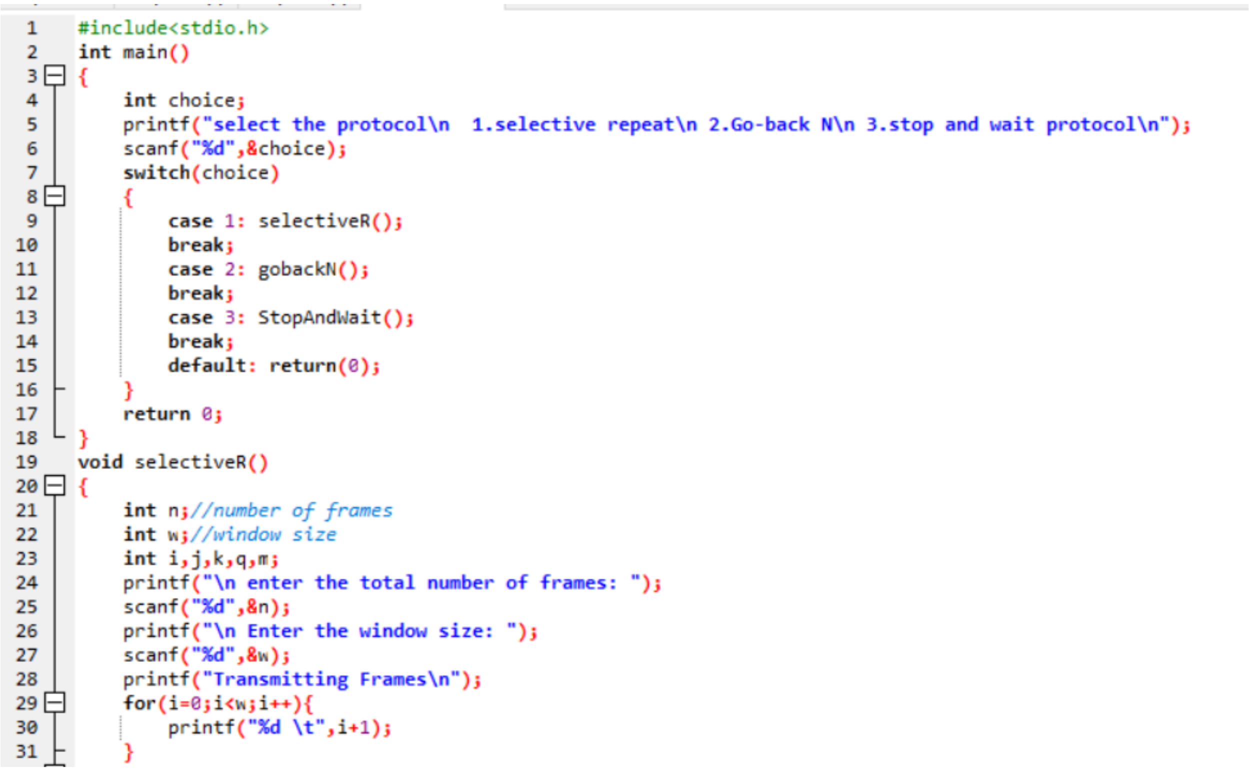
3. Selective Repeat

Start

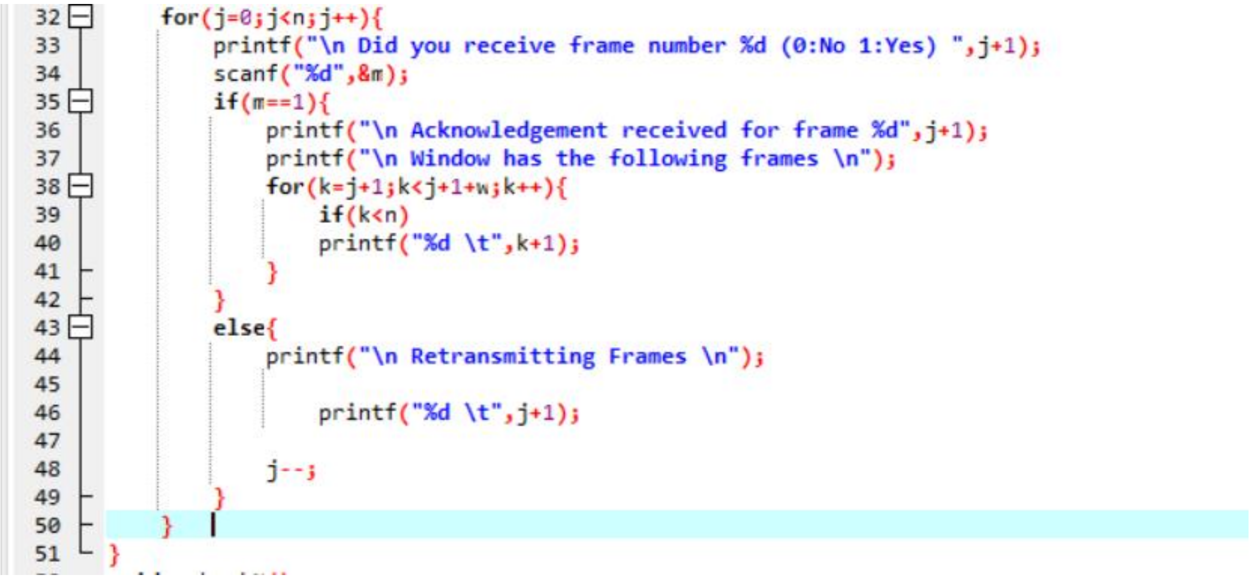
- Wait for frame arrivals.
- Accept any frames in any order and store in a buffer, in the slot corresponding to the sequence number.
- Send negative acknowledgements only for incorrectly frames.
- Advance the window when all slots in the buffer are filled, meaning all frames of that window have been received.

Stop

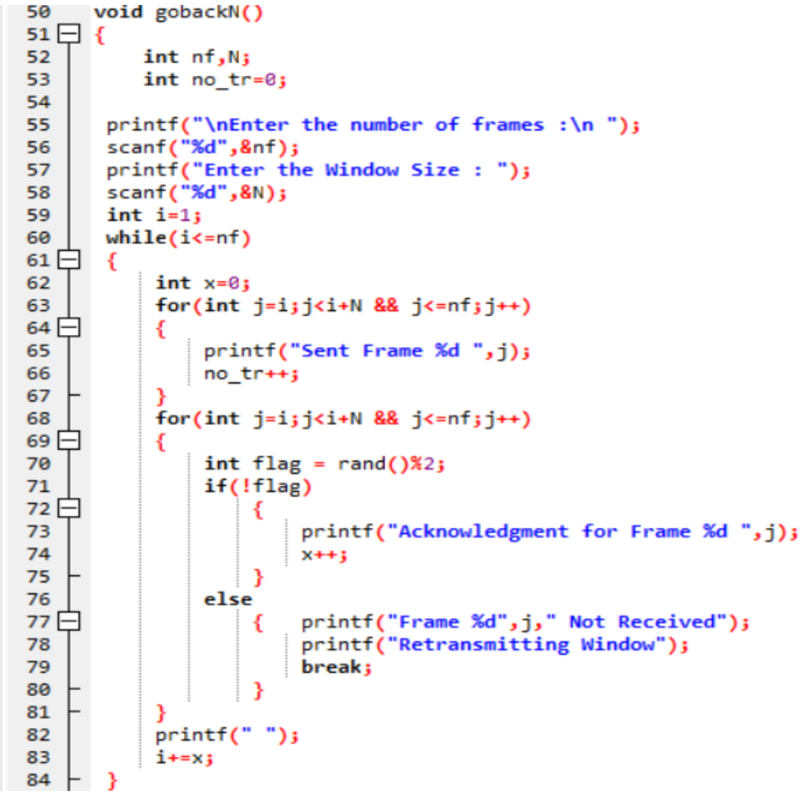
2. Program



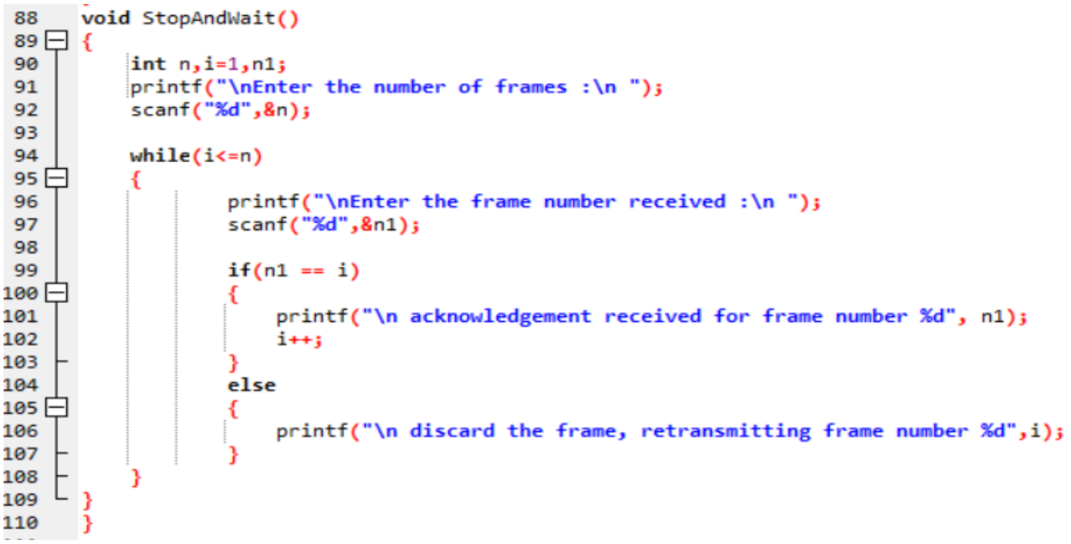
```
#include<stdio.h>
int main()
{
    int choice;
    printf("select the protocol\n  1.selective repeat\n 2.Go-back N\n 3.stop and wait protocol\n");
    scanf("%d",&choice);
    switch(choice)
    {
        case 1: selectiveR();
        break;
        case 2: gobackN();
        break;
        case 3: StopAndWait();
        break;
        default: return(0);
    }
    return 0;
}
void selectiveR()
{
    int n;//number of frames
    int w;//window size
    int i,j,k,q,m;
    printf("\n enter the total number of frames: ");
    scanf("%d",&n);
    printf("\n Enter the window size: ");
    scanf("%d",&w);
    printf("Transmitting Frames\n");
    for(i=0;i<w;i++){
        printf("%d \t",i+1);
    }
```



```
    for(j=0;j<n;j++){
        printf("\n Did you receive frame number %d (0:No 1:Yes) ",j+1);
        scanf("%d",&m);
        if(m==1){
            printf("\n Acknowledgement received for frame %d",j+1);
            printf("\n Window has the following frames \n");
            for(k=j+1;k<j+1+w;k++){
                if(k<n)
                printf("%d \t",k+1);
            }
        }
        else{
            printf("\n Retransmitting Frames \n");

                printf("%d \t",j+1);

            j--;
        }
    }
}
```



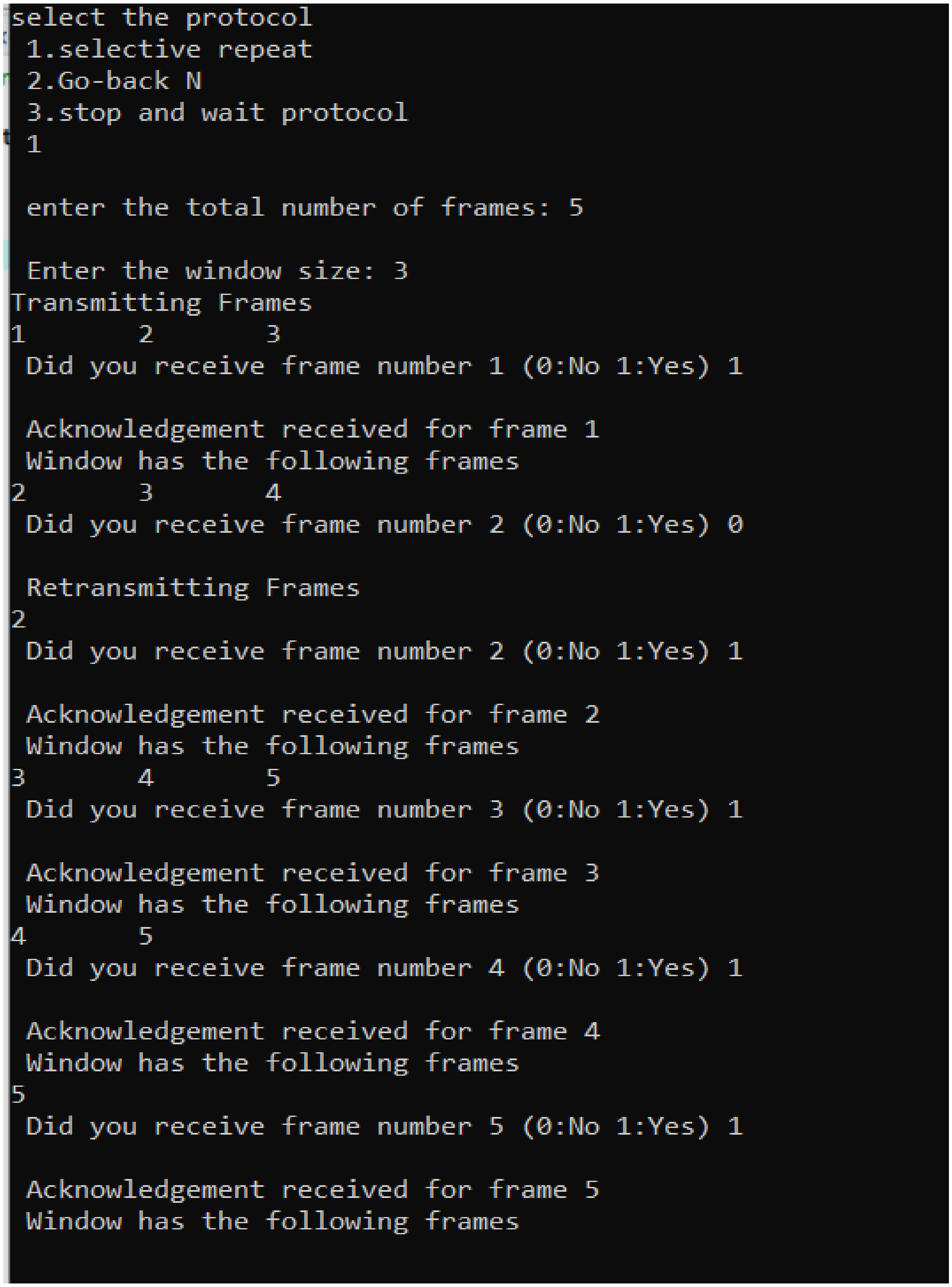
```
void gobackN()
{
    int nf,N;
    int no_tr=0;

 printf("\nEnter the number of frames :\n ");
 scanf("%d",&nf);
 printf("Enter the Window Size : ");
 scanf("%d",&N);
 int i=1;
 while(i<=nf)
 {
     int x=0;
     for(int j=i;j<i+N && j<=nf;j++)
     {
         printf("Sent Frame %d ",j);
         no_tr++;
     }
     for(int j=i;j<i+N && j<=nf;j++)
     {
         int flag = rand()%2;
         if(!flag)
             {
                 printf("Acknowledgment for Frame %d ",j);
                 x++;
             }
         else
             {   printf("Frame %d",j," Not Received");
                 printf("Retransmitting Window");
                 break;
             }
     }
     printf(" ");
     i+=x;
 }
```



```
void StopAndWait()
{
    int n,i=1,n1;
    printf("\nEnter the number of frames :\n ");
    scanf("%d",&n);

    while(i<=n)
    {
            printf("\nEnter the frame number received :\n ");
            scanf("%d",&n1);

            if(n1 == i)
            {
                printf("\n acknowledgement received for frame number %d", n1);
                i++;
            }
            else
            {
                printf("\n discard the frame, retransmitting frame number %d",i);
            }
    }
}
}
```

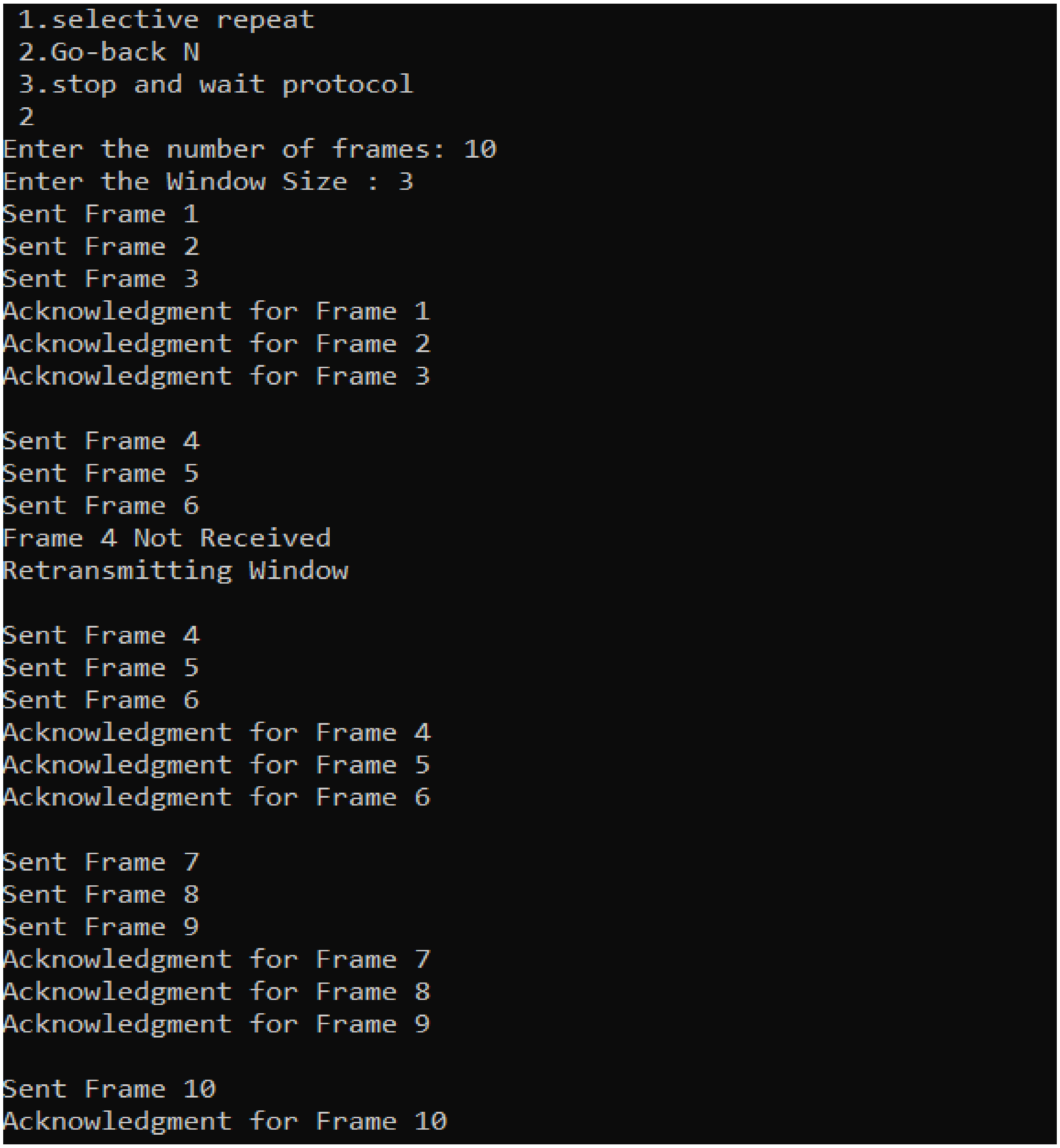
Figure: program to implement stop and wait, go back N and selective repeat ARQ mechanisms.

3. Results



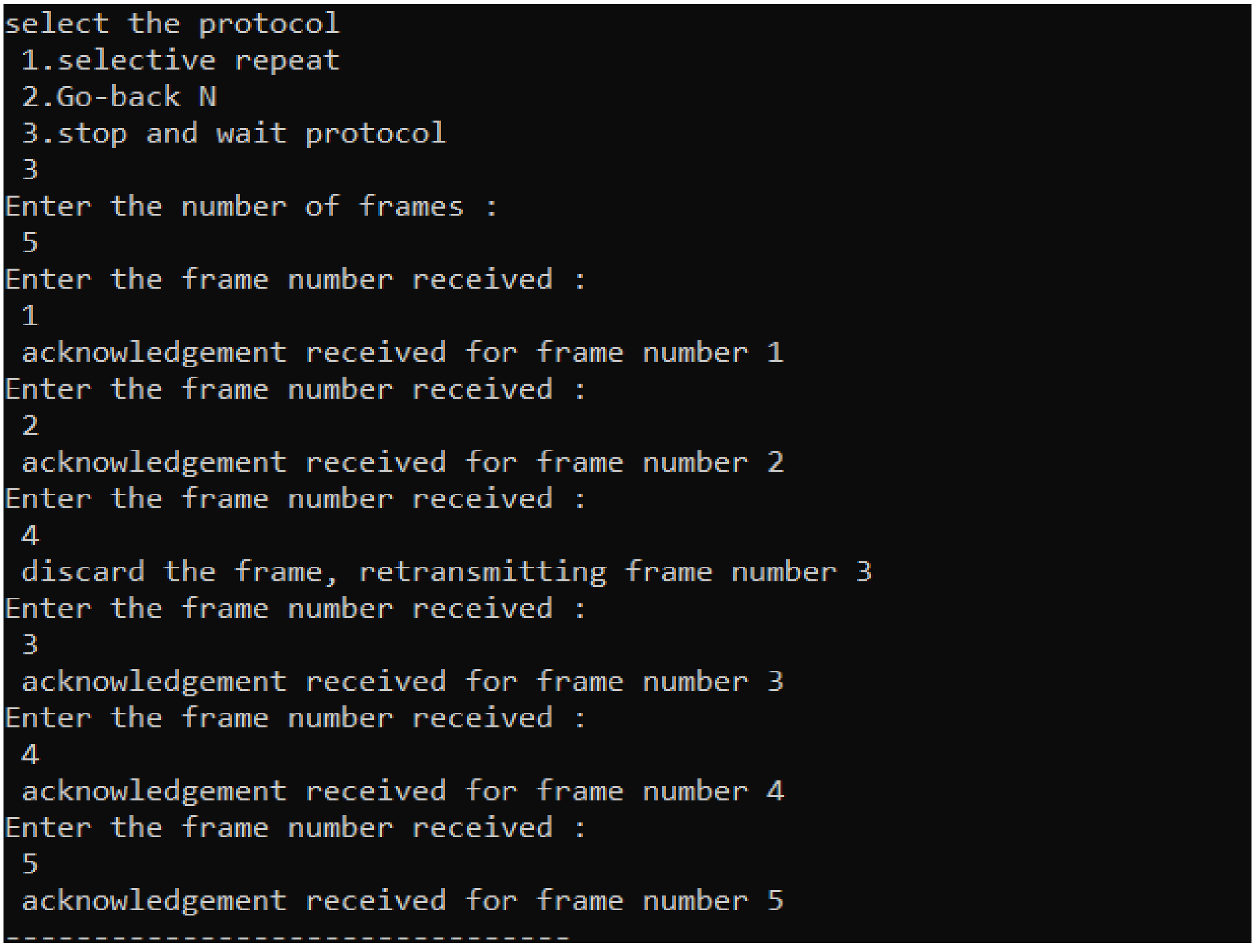
```
select the protocol
 1.selective repeat
 2.Go-back N
 3.stop and wait protocol
 1

 enter the total number of frames: 5

 Enter the window size: 3
Transmitting Frames
1       2       3
 Did you receive frame number 1 (0:No 1:Yes) 1

 Acknowledgement received for frame 1
 Window has the following frames
2       3       4
 Did you receive frame number 2 (0:No 1:Yes) 0

 Retransmitting Frames
2
 Did you receive frame number 2 (0:No 1:Yes) 1

 Acknowledgement received for frame 2
 Window has the following frames
3       4       5
 Did you receive frame number 3 (0:No 1:Yes) 1

 Acknowledgement received for frame 3
 Window has the following frames
4       5
 Did you receive frame number 4 (0:No 1:Yes) 1

 Acknowledgement received for frame 4
 Window has the following frames
5
 Did you receive frame number 5 (0:No 1:Yes) 1

 Acknowledgement received for frame 5
 Window has the following frames
```

Figure: output for selective repeat



```
 1.selective repeat
 2.Go-back N
 3.stop and wait protocol
 2
Enter the number of frames: 10
Enter the Window Size : 3
Sent Frame 1
Sent Frame 2
Sent Frame 3
Acknowledgment for Frame 1
Acknowledgment for Frame 2
Acknowledgment for Frame 3

Sent Frame 4
Sent Frame 5
Sent Frame 6
Frame 4 Not Received
Retransmitting Window

Sent Frame 4
Sent Frame 5
Sent Frame 6
Acknowledgment for Frame 4
Acknowledgment for Frame 5
Acknowledgment for Frame 6

Sent Frame 7
Sent Frame 8
Sent Frame 9
Acknowledgment for Frame 7
Acknowledgment for Frame 8
Acknowledgment for Frame 9

Sent Frame 10
Acknowledgment for Frame 10
```

Figure: output for go back N protocol



```
select the protocol
 1.selective repeat
 2.Go-back N
 3.stop and wait protocol
 3
Enter the number of frames :
 5
Enter the frame number received :
 1
 acknowledgement received for frame number 1
Enter the frame number received :
 2
 acknowledgement received for frame number 2
Enter the frame number received :
 4
 discard the frame, retransmitting frame number 3
Enter the frame number received :
 3
 acknowledgement received for frame number 3
Enter the frame number received :
 4
 acknowledgement received for frame number 4
Enter the frame number received :
 5
 acknowledgement received for frame number 5
--------------------------------
```

Figure: output for stop and wait protocol.

4. Analysis and Discussions
   Difference between stop and wait, go back N and selective repeat.

Sr. No.	Key	Stop and Wait protocol	GoBackN protocol	Selective Repeat protocol
1	Sender window size	In Stop and Wait protocol, Sender window size is 1.	In GoBackN protocol, Sender window size is N.	In Selective Repeat protocol, Sender window size is N.
2	Receiver Window size	In Stop and Wait protocol, Receiver window size is 1.	In GoBackN protocol, Receiver window size is 1.	In Selective Repeat protocol, Receiver window size is N.

3	Minimum Sequence Number	In Stop and Wait protocol, Minimum Sequence Number is 2.	In GoBackN protocol, Minimum Sequence Number is N+1 where N is number of packets sent.	In Selective Repeat protocol, Minimum Sequence Number is 2N where N is number of packets sent.
4	Efficiency	In Stop and Wait protocol, Efficiency formular is 1/(1+2*a) where a is ratio of propagation delay vs transmission delay.	In GoBackN protocol, Efficiency formular is N/(1+2*a) where a is ratio of propagation delay vs transmission delay and N is number of packets sent.	In Selective Repeat protocol, Efficiency formular is N/(1+2*a) where a is ratio of propagation delay vs transmission delay and N is number of packets sent.
5	Acknowledgement Type	In Stop and Wait protocol, Acknowledgement type is individual.	In GoBackN protocol, Acknowledgement type is cumulative.	In Selective Repeat protocol, Acknowledgement type is individual.
6	Supported Order	In Stop and Wait protocol, no specific order is needed at receiver end.	In GoBackN protocol, in-order delivery only are accepted at receiver end.	In Selective Repeat protocol, out-of-order deliveries also can be accepted at receiver end.
7	Retransmissions	In Stop and Wait protocol, in case of packet drop,number of retransmition is 1.	In GoBackN protocol, in case of packet drop,numbers of retransmitions are N.	In Selective Repeat protocol, in case of packet drop,number of retransmition is 1.

5. Conclusions

The selective repeat is a more efficient protocol than go back N as it does not waste bandwidth for the frames which are properly received but, its complexity and expense favors the use of the go-back-n protocol. Go-Back-*N* ARQ is a more efficient use of a connection than Stop-and-wait ARQ, since unlike waiting for an acknowledgement for each packet, the connection is still being utilized as packets are being sent. Comments

a. Limitations of the experiment and result

limitations of stop-and-wait:

- fairly slow: the sender can send at most one new packet per RTT.
- not robust: if the ack can get lost, when the receiver gets a packet, the receiver cannot tell if it is a retransmission or a new packet.

Limitations of go back n:

- not efficient
- waste bandwidth when a packet is lost/broken.

Limitations of selective repeat:

- complicated with multiple timers
- receiver needs a buffer to buffer out-of-order packets.

b. Learning

Learnt programs to demonstrate stop and wait, go back N and selective repeat ARQ mechanisms. We learnt their advantages and disadvantages and efficiencies.

c. Recommendations: none

© Ramaiah University of Applied Sciences