

**Ex: 6.b****GO BACK N ARQ**

Implement Go Back n ARQ using socket programming.

Sender should perform the following:

1. Read the contents from the user.
2. Divide the contents into n frames.
3. Set the window size
4. Number the frames sequentially and find the parity check.
5. Add the header and trailer accordingly.
6. Send the frames one by one.
7. When the sender receives an ack by time, slide the window accordingly. Else retransmit all the frames numbered from the previous ack (for time out condition).
8. Also consider for lost of delayed ack.

Receiver should do the following:

1. Receive the frames from the sender.
2. Verify the sequence number of the frames to identify whether the receiver is receiving the frame in order.
3. If received a frame, check whether the sequence number of the frame is same as the frame number expected in window. If not, discard the frame.
4. If received properly slide the receiver window.
5. Receiver sends ack randomly.

**Sample Input Output**

**Note:** Sample input and output is without header and trailer. When you do the program insert it properly.

**Receiver**

Frame 0 – 01010101

The data received is correct

Do you want to send ack

1.yes

2.no

1

Frame 1 – 0100000

Error in frame

Do you want to send ack

1.yes

2.no

2

**Receive other frames not in window**

**Discard the frame**

Receiving frame 1

Frame 1 – 01000010

Do you want to send ack

1.yes

2.no

.....

**Sender**

Sender window size 2

Data to sent.... Frame 0 – 01010101

Do you want to introduce error?

1.yes

2.no

2

Do the frame gets lost”

1.yes

2.no

2

Frame 0 01010101 sent

Received Ack1

Frame 1 – 01000010

Do you want to introduce error?

1.yes

2.no

1

Introduce error in which position—2

Frame 1 01000000 sent

**Mean while send other frames**

**Time out for Frame 1 ack**

**Resend Frame1**

.....