8/9/25 Practical - 7

AIM:

Write a program to implement flow control at data link layer using sliding window protocol simulate the flow of frames from one node to another.

Sender program:

1) Input window size and text message
2) Create frames [Frame no., DATA]
3) Print + save frames in sender_Buffer
4) Wait (delay). Then read Receiver_Buffer.
5) If ACK = expected → send next frames (overwrite sender_Buffer).
6) If NACK → resend old frames (overwrite sender_Buffer)

Receiver Program:

1) Read sender_Buffer.
2) Check frame numbers.
3) If correct → write ACK in Receiver_Buffer
4) If incorrect → write NACK in Receiver_Buffer

Program:

```
from re import X
import time
import os
os.system('clear')
SB = open ("sender_Buffer.txt", "a+")
RB = open ("Receiver_Buffer.txt", "r+")
SB.truncate (0)
RB.truncate(0)
ws = int (input ("Enter window size:"))
s = input ("Enter input string:")
s = list (s)
if (ws < len(s)):
    for i in range (0, len(s), ws):
        p = s[i: i+ws]
        y = s[i + ws: i+ws+ws]
```

```
        Print ("Sent → " + str(p))
        time.sleep(ws)
        Print ("Sending → ", str(y))
    x = 0
    while (x < ws):
        time.sleep(2)
        if (len(p) > x):
            Print ("ACK ~! ", P[x], " !")
            RB.write (P[x])
        time.sleep(1)
        if (len(y) > x):
            print ("Sending → ", y[x])
            SB.write (y[x])
        x += 1
    else:
        print ("~> The window size is too large.")
```

SAMPLE INPUT OUTPUT:

```
Enter window size : 3
Enter Input String : HELLO WORLD
Sent → ['H', 'E', 'L']
Sending → ['L', 'O', 'W']
ACK ~! H !
ACK ~! E !
ACK ~! L !
Sending → L
Sending → O
Sending → W
Sent → ['O', 'R', 'L']
Sending → ['D']
ACK ~! O !
ACK ~! R !
ACK ~! L !
Sending → D
```

Result :

Hence the required data is sent using Sliding window

19/8/25