CS23532-COMPUTER NETWORKS-LAB MANUAL
Practical -7

Name: Tharunraj
RegNo:230701362

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.

Program should achieve at least below given requirements. You can make it a bidirectional program wherein receiver is sending its data frames with acknowledgement (Piggybacking)

Create a sender program with following features:-

1. Input Window size from the user.
2. Input a Text message from the user.
3. Consider 1 character per frame.
4. Create a frame with following fields [Frame no., DATA].
5. Send the frames. [Print the output on screen and save it in a file called Sender_Buffer.]
6. Wait for the acknowledgement from the Receiver. [Induce delay in the program]
7. Reader a file called Receiver_Buffer.
8. Check ACK field for the Acknowledgement number.
9. If the Acknowledgement number is as expected, send new set of frames accordingly, [overwrite the Sender_Buffer file with new frames] Else if NACK is received, resend the frames accordingly. [Overwrite the Sender_Buffer with old frame].

Create a receiver file with following features

1. Reader a file called Sender_Buffer.
2. Check the Frame no.
3. If the Fame no. are as expected, write the appropriate ACK no. in the Receiver_ Buffer file. Else write NACK no. in the Receiver_Buffer file.

NOTE: Induce error and verify the behaviour of the program. Manually Change the Frame no and Ack no in the files].

Program:

Sender.py

```
import time

win = int(input("Enter window size: "))
msg = input("Enter message: ")

frames = [[i + 1, msg[i]] for i in range(len(msg))]
ack = 0

while ack < len(frames):
    send = frames[ack:ack + win]
    print("\nSending frames:", send)
    with open("Sender_Buffer.txt", "w") as f:
        for fr in send:
            f.write(f"{fr[0]} {fr[1]}\n")
    time.sleep(1)  # simulate delay

    with open("Receiver_Buffer.txt") as f:
        res = f.read().strip()
    if res.startswith("ACK"):
        ack_no = int(res.split()[1])
        print("Received:", res)
        ack = ack_no
    else:
        print("Received:", res, "→ Resending same frames.")
print("\nAll frames sent successfully!")
```

receiver.py

```
try:
    with open("Sender_Buffer.txt") as f:
        lines = [line.strip().split() for line in f.readlines()]
except FileNotFoundError:
    exit("No sender buffer found!")

expected = 1
valid = True
for fr_no, data in lines:
    if int(fr_no) != expected:
        valid = False
        break
    expected += 1

with open("Receiver_Buffer.txt", "w") as f:
    if valid:
        f.write(f"ACK {expected-1}")
        print("Frames received correctly. Sending ACK", expected-1)
    else:
```

```
        f.write(f"NACK {expected}")
        print("Frame error. Sending NACK", expected)
```

Input:

Enter window size: 3
Enter message: HELLO

Output:

Sending frames: [[1, 'H'], [2, 'E'], [3, 'L']]
Received: ACK 3

Sending frames: [[4, 'L'], [5, 'O']]
Received: ACK 5

All frames sent successfully!

RESULT:

Thus the program to implement flow control at data link layer using sliding window protocol has been executed successfully.