14/9/24 PRACTICAL - 7

AIM: Write a program to implement flow control at data link layer using sliding window protocol. Simulate the flow of frames frome one node to another.

PROGRAM:

```
import threading.
import time.
import random.
import queue.

sender_to_reciever_queue = queue.Queue()
reciever_to_sender_queue = queue.Queue()

def sender (window_size message):
    frames = [message [i:i+1] for i in
range (0, len (message ))]
num_frames = len(frames)
base = 0
next_seq_num = 0
while base < num_frames:
    print (f" window Base: {base}")
for i in range (base, min (base + window_size
    . num_frames)):
    frame = (i, frames[i])
    sender_to_receiver_queue.put (frame)
```

```
print(f" sending frame no: {i} ,DATA: {frames[i]}")
time.sleep(2)
ack_received = set()
while not receiver_to_sender_queue.empty():
    ack = receiver_to_sender_queue.get()
    if ack.startswith('ACK'):
        ack_num = int(ack.split(':')[1])
        ack_received.add(ack_num)
    elif ack.startswith('NACK'):
        nack_num = int(ack.split(":")[1])
print(f" received NACK for frame no: {nack_num}.
    resending frames starting from {nack_num}:")
base = nack_num
next_seq_num = base.
if base >= num_frames:
    print(" All frames successfully sent and
                acknowledged.")
    break.
```

```
def receiver():
    expected_frame_no=0
while True:
    try:
        frame_no, data = sender_to_receiver_
                queue.get(timeout=5).
```

```
except queue.Empty:
    break
if random.random() > 0.1:
    ack = f"ACK: {frame_no}"
    print(f"received frame no: {frame_no}
    sending ACK")
receiver_to_sender_queue.put(ack)
    if frame_no == expected_frame_no:
        expect_frame_no += 1
else:
    ack = f"NACK: {frame_no}"
    print(f"received frame no: {frame_no}
, sending NACK due to error.")
    receiver_to_sender_queue.put(ack).
    reciever_to_sender_queue..put("END")

def main():
    window_size = int(input("Enter window size:"))
message = input("Enter text message:")
receiver_thread = threading.Thread(target=
                                   receiver)
    receiver_thread.start()
    sender(window_size, message)
receiver_thread.join()
if __name__ == "__main__":
    main().
```

Output

Python sender.py

Enter window size : 3

Enter text message = hello

Sending frames [(0,'h'),(1,'e'),(2,'l')]

Ack recieved for frame 1

Python reciever.py

No. of frames, No. process - waiting -

Sending ACK

Received frame 0 : h

Sending ACK.

Received frame 1 : e

Sending ACK.

Received frame 2 : l

Sending ACK.

Received frame 3 : l

Sending ACK.

Recieved frame 4 : o

Sending.

Result The output for sliding window is successfully executed and output is verified.