CN - Lab-Test - 2

VYSHNAVI.M.N
1BM18CS125

4) Design an algorithm for congestion control setting output rate and buffer size. Show successful transmission of packets along with number of bytes transmitted. Check for overflow condition.

```
class LeakyBucket:
    def __init__(self, bucket_size, input_stream,
                     output_rate):

        self.size = bucket_size
        self.queue = input_stream
        self.flow = output_rate

  def control_congestion(self):
     buffer = 0
     for packet in self.queue:
         print(f"Incoming packet: {packet}")
         x = self.size - buffer
         if packet < x:
             buffer += packet
             print(f"Packets sent: {self.flow}\n
                     Buffer: {buffer} \n")
           if buffer > self.size: print("Overflow.")
       else:
          print(f"Packets loss: {packet - x} \t
                  Packets sent: {self.flow}")
```

1BM18CS125
VYSHNAVI.M.N

```
    buffer = self.size
    print(f"Buffer : {buffer} \n")
  buffer = buffer - self.flow
  printf(f"Buffer : {buffer} \t Packets Sent :
          {self.flow}")
  while buffer :
       sent = self.flow if self.flow < buffer
              else buffer
       buffer = buffer - sent
       print(f"Buffer : {buffer} \t Packets Sent :
              {sent}")

input_stream = [int(x) for x in input("Enter
               the input stream of packets : ").
               split(' ')]
bucket_size = int(input("Enter bucket size : "))
output_rate = int(input("Enter output data
                          rate "))
network = LeakyBucket(bucket_size, input_strea
                      output_rate)
network.control_congestion()
```