Date __/__/__
Page _____

3. Write a program for congestion control using Leaky Bucket algorithm.

Sol.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#define NOF_PACKETS 5

int rand(int a){
    int rn = (random() % 10) % a;
    return rn == 0 ? 1 : rn;
}

int main(){
    int packet_sz[NOF_PACKETS], i, clk, b_size,
    o_rate, p_sz_rm=0, p_sz, p_time, op;

    for(i=0; i<NOF_PACKETS; i++)
        packet_sz[i] = random() % 100;
    for(i=0; i<NOF_PACKETS; i++)
        printf("\n packet[%d]: %d bytes \t", i, packet_sz[i]);

    printf("Enter Output rate");
    scanf("%d", &o_rate);
    printf("%d", &b_size);
    for(i=0; i<NOF_PACKETS; i++)
        if(packet_sz[i] + p_sz_rm) > b_size)
            if(packet_sz[i] > b_size)
                printf("Incoming packet size (%d bytes) is
                    greater than bucket capacity (%d bytes)
                    -PACKET REJECTED", packet_sz[i],
                    b_size);
            else
                printf("\n Bucket capacity exceeded!");
```

Date ___/___/___
Page _____

```
else {
    p_sz_rm = packet_sz[i];
    printf("Size", packet_sz[i]);
    printf("Bytes remaining to Transmit:", p_sz_rm);
while (p_sz_rm > 0) {
    sleep(1);
    if (p_sz_rm) {
     if (p_sz_rm <= o_rate)
      op = p_sz_rm, p_sz_rm = 0;
     else
      op = o_rate, p_sz_rm -= o_rate;

      printf(op)
      printf("Bytes remaining:", p_sz_rm);
    }
    else {
      printf("No packets to transmit!");
    }
  }
 }
}
```

Output:

packet[0]: 83 bytes
packet[1]: 86 bytes
packet[2]: 77 bytes
Enter output rate: 30
Enter Bucket size: 85

Incoming packet size: 83
Bytes remaining to Transmit: 83