

CYCLE -2 LEAKYBUCKET

1. Write a program for congestion control using CRC- CCITT (16-bit).

leakybucket
1) write a program for congestion control using leaky bucket algorithm.

```
program : lbucket.cc
#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;
}
*/
/*
#include <stdlib.h>
long int random(void);
*/
int main() {
    int packet_sz[NOF_PACKETS], i, clk, b_size, o_rate,
    p_sz, rtm = 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("\n Enter the Output rate:");
    scanf("%d", &o_rate);
    printf("\n Enter the bucket size:");
    scanf("%d", &b_size);
    for (i = 0; i < NOF_PACKETS; ++i)
```

```

{ if (packet_sz[i] + p_sz_rem) > b_size)
    if (packet_sz[i] > b_size) {
        /* Compare the packet size with bucket size */
        printf("\n\n Incoming packet size (%d bytes) is
        Greater than bucket capacity (%d bytes) - PACKET
        REJECTED ", packet_sz[i], b_size);
        else:
            printf("\n\n Bucket capacity exceeded - PACKETS
            REJECTED!!");
        else {
            p_sz_rem += packet_sz[i];
            printf("\n\n Incoming packet size: %d", packet_sz[i]);
            printf("\n\n Bytes remaining to transmit: %d", p_sz_rem);
            // p_time = rand() * 10;
            // printf("\n\n Time left for transmission: %d units",
            p_time);
            // for (clk = 10; clk <= p_time; clk += 10)
            while (p_sz_rem > 0) {
                sleep(1);
                if (p_sz_rem) {
                    if (p_sz_rem <= 0.9 * rate) /* packet size remaining
                    Comparing with output rate */
                        op = p_sz_rem, p_sz_rem = 0;
                    else
                        op = 0.9 * rate, p_sz_rem -= 0.9 * rate;
                    printf("\n\n Packet of size %d Transmitted", op);
                    printf("\n\n --- Bytes Remaining to Transmit: %d",
                    p_sz_rem);
                }
            }
            else {
                printf("\n\n No packets to transmit!!"); } } } }

```


Output:

packet[0]: 83 bytes

packet[1]: 86 bytes

packet[2]: 77 bytes

packet[3]: 15 bytes

packet[4]: 93 bytes

Enter the output rate: 30

Enter the bucket size: 85

Incoming packet size: 83

Bytes remaining to transmit: 83

packet of size 30 Transmitted --- Bytes Remaining to transmit: 53

packet of size 30 Transmitted --- Bytes Remaining to transmit: 23

packet of size 23 Transmitted --- Bytes Remaining to transmit: 0

Incoming packet size (86 bytes) is greater than bucket capacity (85 bytes) - packet Rejected.

Incoming packet size: 77

Bytes remaining to transmit: 77

packet of size 30 Transmitted → Bytes Remaining to transmit: 47

packet of size 30 Transmitted → Bytes Remaining to transmit: 17

packet of size 17 Transmitted → Bytes Remaining to transmit: 0

Incoming packet size: 15

Bytes remaining to transmit: 15

packet of size 15 Transmitted → Bytes Remaining to transmit: 0

OUTPUT:

```
packet[0]:83 bytes
packet[1]:86 bytes
packet[2]:77 bytes
packet[3]:15 bytes
packet[4]:93 bytes
Enter the Output rate:30
Enter the Bucket Size:85

Incoming Packet size: 83
Bytes remaining to Transmit: 83
Packet of size 30 Transmitted----Bytes Remaining to Transmit: 53
Packet of size 30 Transmitted----Bytes Remaining to Transmit: 23
Packet of size 23 Transmitted----Bytes Remaining to Transmit: 0

Incoming packet size (86bytes) is Greater than bucket capacity (85bytes)-PACKET REJECTED

Incoming Packet size: 77
Bytes remaining to Transmit: 77
Packet of size 30 Transmitted----Bytes Remaining to Transmit: 47
Packet of size 30 Transmitted----Bytes Remaining to Transmit: 17
Packet of size 17 Transmitted----Bytes Remaining to Transmit: 0

Incoming Packet size: 15
Bytes remaining to Transmit: 15
Packet of size 15 Transmitted----Bytes Remaining to Transmit: 0
```