Practical 15

Congestion control protocols : Token Bucket

* Objective

To simulate the continuous working of the Leaky Bucket algoritm for managing network traffic by regulating packet flow and preventing overflow.

* Code

```
#include <stdio.h>
#include<stdbool.h>
#include <unistd.h>

int main() {
    int bucket_size,output_rate;

    printf("Enter the bucket size: ");
    scanf("%d", &bucket_size);
    printf("Enter the output rate of the
            bucket: ");
    scanf("%d", &output_rate);

    int bucket = 0;

    while(true)
    {
        int incoming_packets;
        printf("Enter the number of incoming packets: ");
        scanf("%d", &incoming_packets);
```

```
if (bucket + incoming_packets <= bucket_size)
{
    bucket += incoming_packets;
}
else
{
    printf("Bucket overflow! Dropping %d
            packets.\n", incoming_packets +
            bucket - bucket_size);
    bucket = bucket_size;
}
if (bucket >= output_rate)
{
    printf("%d packets transmitted.\n",
            output_rate);
    bucket -= output_rate;
}
else
{
    printf("Bucket empty.\n");
}
    usleep(1000000);
}
return 0;
}
```

* # VRC

```
#include <stdio.h>

int main()
{
    int data[10], vrc = 0;

    printf("Enter 7 bits of data: ");
    for(int i = 0; i < 7; i++)
    {
        scanf("%d", &data[i]);
    }
    for(int i = 0; i < 7; i++)
    {
        vrc ^= data[i];
    }

    printf("VRC bit is: %o\n", vrc);

    return 0;
}
```

* Output

```
Enter the bucket size: 10
Enter the output rate of the bucket: 4
4 packets transmitted
Enter the number of incoming packets: 7
4 packets transmitted
```

* Output

```
Enter 7 bits of data : 1
0
1
0
1
0
1
VRC bit is : 0
```

* Learning Outcome

Understand and implement leaky Bucket and VRC (Vertical Redundancy Check) for traffic control and error detection.