WEEK 14

Write a program for congestion control using Leaky bucket algorithm.

CODE:
```c
#include <stdio.h>
#include <stdlib.h> // Include this for the rand() function
int main()
{
    int buckets, outlets, k = 1, num, remaining;
    printf("Enter Bucket size and outstream size\n");
    scanf("%d %d", &buckets, &outlets);
    remaining = buckets;
    while (k)
    {
        num = rand() % 1000; // Generate a random number between 0 and 999
        if (num < remaining)
        {
            remaining = remaining - num;
            printf("Packet of %d bytes accepted\n", num); // Added missing variable
        }
        else
        {
            printf("Packet of %d bytes is discarded\n", num);
        }
        if (buckets - remaining > outlets)
        {
            remaining += outlets; // Fixed the calculation
        }
        else
            remaining = buckets;
        printf("Remaining bytes: %d \n", remaining);
```

```c
        printf("If you want to stop input, press 0, otherwise, press 1\n");
        scanf("%d", &k);
    }
    while (remaining < buckets) // Fixed the condition
    {
        if (buckets - remaining > outlets)
        {
            remaining += outlets; // Fixed the calculation
        }
        else
            remaining = buckets;
        printf("Remaining bytes: %d \n", remaining);
    }
    return 0; // Added a return statement to indicate successful completion
}
```

OUTPUT:

```
PS D:\VS Code> cd "d:\VS Code\OS\" ; if ($?) { gcc bucket.c -o bucket } ; if ($?) { .\bucket }
Enter Bucket size and outstream size
2000
100
Packet of 41 bytes accepted
Remaining bytes: 2000
If you want to stop input, press 0, otherwise, press 1
1
Packet of 467 bytes accepted
Remaining bytes: 1633
If you want to stop input, press 0, otherwise, press 1
1
Packet of 334 bytes accepted
Remaining bytes: 1399
If you want to stop input, press 0, otherwise, press 1
1
Packet of 500 bytes accepted
Remaining bytes: 999
If you want to stop input, press 0, otherwise, press 1
1
Packet of 169 bytes accepted
Remaining bytes: 930
If you want to stop input, press 0, otherwise, press 1
1
Packet of 724 bytes accepted
Remaining bytes: 306
If you want to stop input, press 0, otherwise, press 1
1
Packet of 478 bytes is discarded
Remaining bytes: 406
If you want to stop input, press 0, otherwise, press 1
1
Packet of 358 bytes accepted
Remaining bytes: 148
If you want to stop input, press 0, otherwise, press 1
1
Packet of 962 bytes is discarded
Remaining bytes: 248
If you want to stop input, press 0, otherwise, press 1
0
Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
```

```
Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
Remaining bytes: 848
Remaining bytes: 948
Remaining bytes: 1048
Remaining bytes: 1148
Remaining bytes: 1248
Remaining bytes: 1348
Remaining bytes: 1448
Remaining bytes: 1548
Remaining bytes: 1648
Remaining bytes: 1748
Remaining bytes: 1848
Remaining bytes: 1948
Remaining bytes: 2000
PS D:\VS Code\OS> []
```

OBSERVATION:

Experiment -14

Date 17/8/23
Page _____

Aim: To write a program for congestion control using leaky bucket algorithm.

```
#include <stdio.h>
#include <conio.h>

void main() {
    int bucket-size, dr;
    printf("Enter bucket size and data state");
    scanf("%d %d ", &bucket-size, &dr);
    int temp = bucket-size;
    while(1){
        int ch, ps;
        printf("Enter packet size\n");
        scanf("%d", &ps);
        if ( ps <= bucket - size ) {
            if (ps <= temp)
                printf("packet size %d sent", ps);
            else {
                printf("packet size %d dropped", ps);
                ps=0;
            }
            temp = temp - ps + dr;
        }
        else
            printf("packet dropped");
        printf("continue transmission ?");
        scanf("%d", &ch);
        if (ch==0)
            break;
    }
}
```

Output :

Enter bucketsize and data erate
4000   250
Enter packet size
5000

packet dropped
continue transmission?

1

Enter packet size
1000
packet size 1000 sent
continue transmission?

1

Enter packet size
3000

packet size 3000 sent
continue transmission?

1

Enter packet size
750
packet size 750 dropped
continue transmission?

0