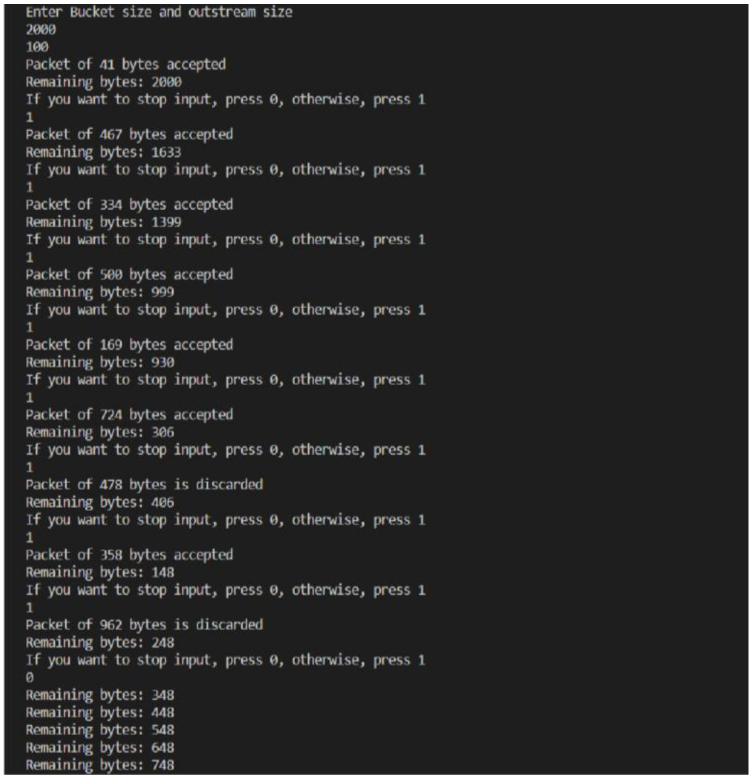
LAB 14

Write a program for congestion control using Leaky bucket algorithm.

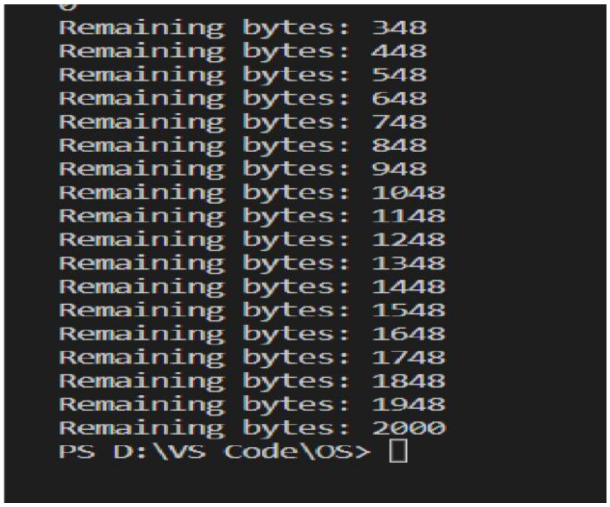
CODE:

```
#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); 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:

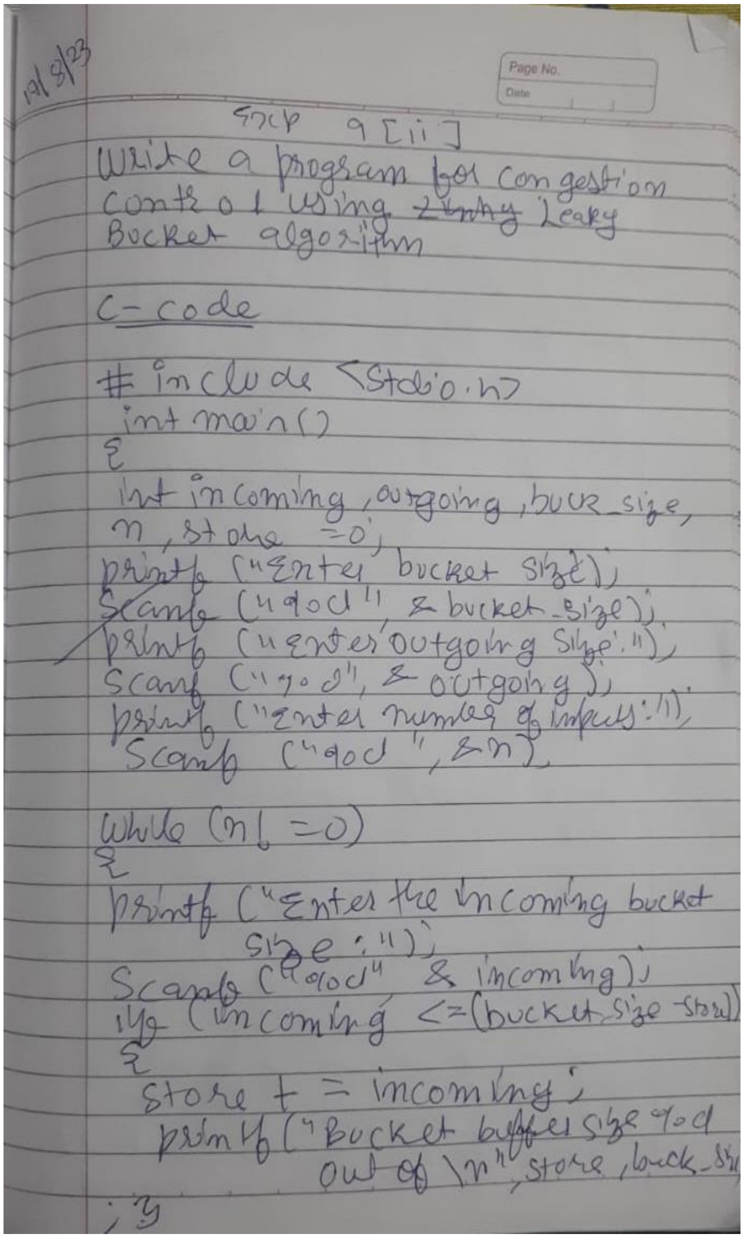


```
Enter Bucket size and outstream size
2000
100
Packet of 41 bytes accepted
Remaining bytes: 2000
If you want to stop input, press 0, otherwise, press 1
1
Packet of 467 bytes accepted
Remaining bytes: 1633
If you want to stop input, press 0, otherwise, press 1
1
Packet of 334 bytes accepted
Remaining 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:



12/8/23

Page No.
Date

Step 9 [ii]

Write a program for congestion control using ~~[illegible]~~ Leaky Bucket algorithm

C-code

```
# include <Stdio.h>
 int main()
{
 int incoming, outgoing, buck_size,
 n, store =0;
printf ("Enter bucket size");
Scanf ("%d", & bucket_size);
printf ("Enter outgoing size:");
Scanf ("%d", & outgoing);
printf ("Enter number of inputs:");
 Scanf ("%d", &n);

While (n! =0)
{
printf ("Enter the incoming bucket
          size:");
Scanf ("%d" & incoming);
if (incoming <=(bucket_size-store))
 {
  store + = incoming;
   printf ("Bucket buffer size %d
            out of \n", store, buck_si
;}
```

Page No.
Date

```
else
{
printf("Dropped %d no of packets\n"
        incoming -(buck_size-store));

printf("Bucket buffer size %d out of
        %d \n", store, buck_size);
store = buck_size;
}
store = store - outgoing;
printf (" After outgoing %d packets
        left out of %d in buffer\n",
        store, buck_size);
   n--;
  }
}
```

19/8/2024

Output :-

```
Enter bucket size:500
enter outgoing rate:-2000
enter number of inputs:- 2
enter the incoming packet size!3000
Bucket buffer size 3000 out of 5000
After outgoing 1000 packets left out of
 5000 in buffer.
Enter the incoming packet size:1000
Bucket buffer size 2000 out of 5000
After outgoing 0 packets left
 out of 5000 in buffer.
```