WEEK 14

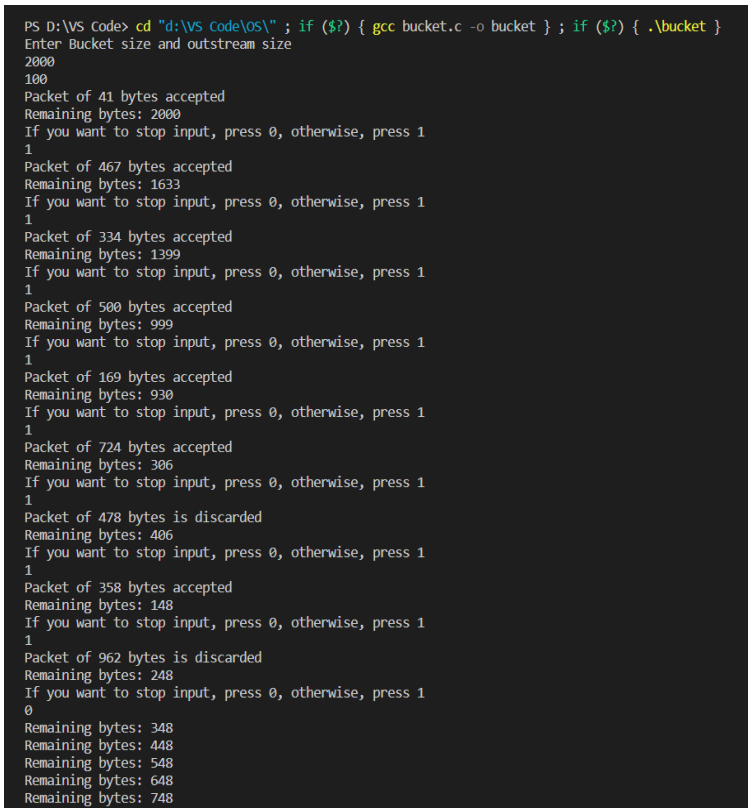
Write a program for congestion control using Leaky bucket algorithm.

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);
```

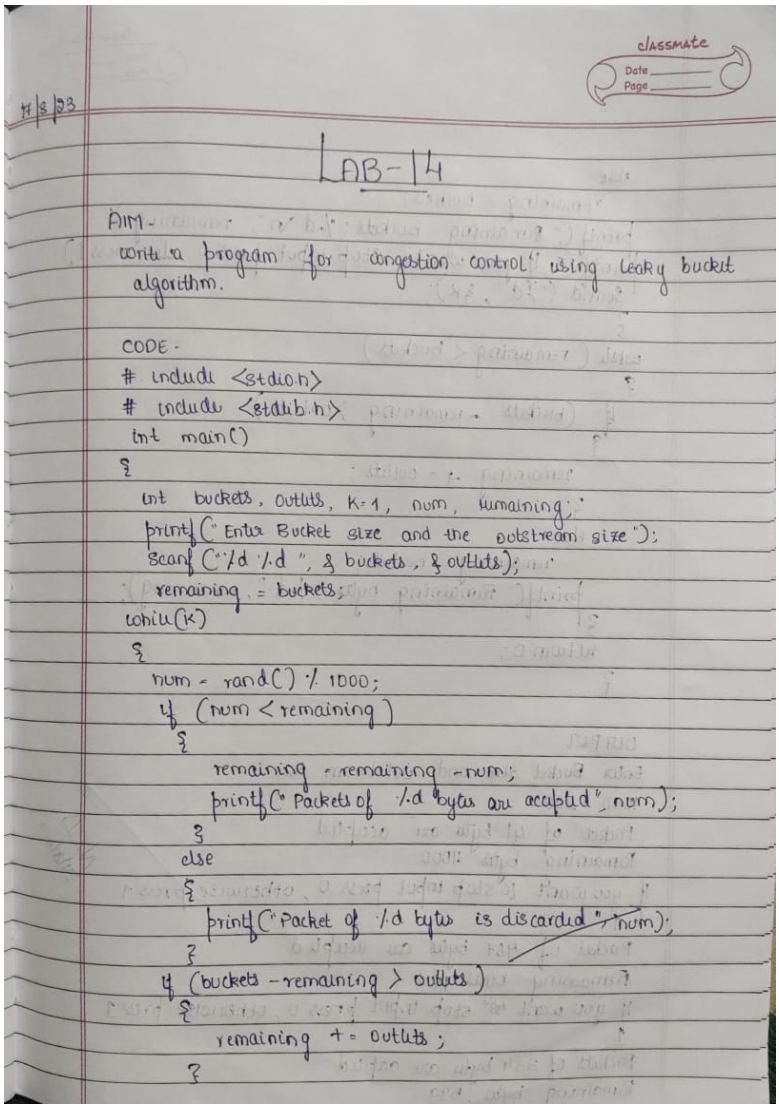
```
        printf("If you want to stop input, press 0, otherwise, press 1\n");
        scanf("%d", &k);
    }
    while (remaining < buckets) // Fixed the condition
    {
        if (buckets - remaining > outlets)
        {
            remaining += outlets; // Fixed the calculation
        }
      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

OUTPUT:



```
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:



LAB-14

AIM -
write a program for congestion control using Leaky bucket algorithm.

CODE -

```
# includu <stdio.h>
# includu <stdlib.h>
int main()
{
  int buckets, outlets, K=1, num, remaining;
  printf("Enter Bucket size and the outstream size");
  scanf("%d %d", &buckets, &outlets);
  remaining = buckets;
while(K)
{
  num = rand() % 1000;
  if (num < remaining)
    {
      remaining = remaining - num;
      printf("Packets of %d bytes are accepted", num);
    }
  else
    {
      printf("Packet of %d bytes is discarded", num);
    }
  if (buckets - remaining > outlets)
    {
      remaining += outlets;
    }
```

```
else
    remaining = buckets;
printf(" Rimaining buckets : %d \n", remaining);
printf(" If you want to stop input press 0, else press 1");
scand ("%d", &x);
}
while (remaining < buckets)
{
  if (buckets - remaining > outlets)
  {
    remaining += outlets;
  }
  else
    remaining = buckets;
  printf(" Rimaining byts : %d ", remaining);
}
  return 0;
}
```

OUTPUT:

Enter Bucket size and outstream size

1000 200

Packet of 41 byts are accepted

Rimaining byts :1000

If you want to stop input press 0, otherwise press 1

1

Packet of 467 byts are accepted

Remaining byts :733

If you want to stop input press 0, otherwise press 1

1

Packets of 334 byts are aapted

Rimaining byts : 599