WEEK 14

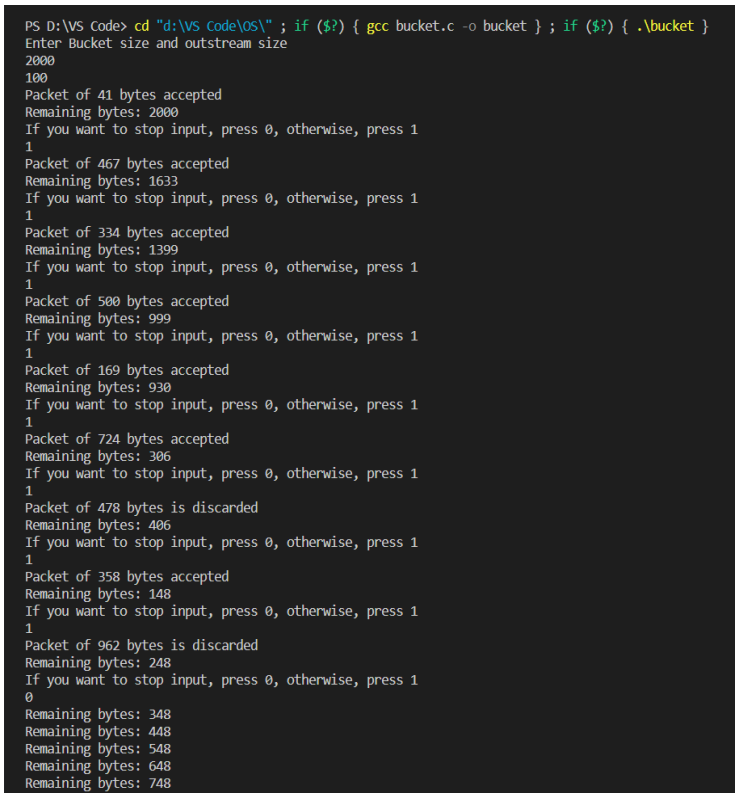
Write a program for congestion control using Leaky bucket algorithm.

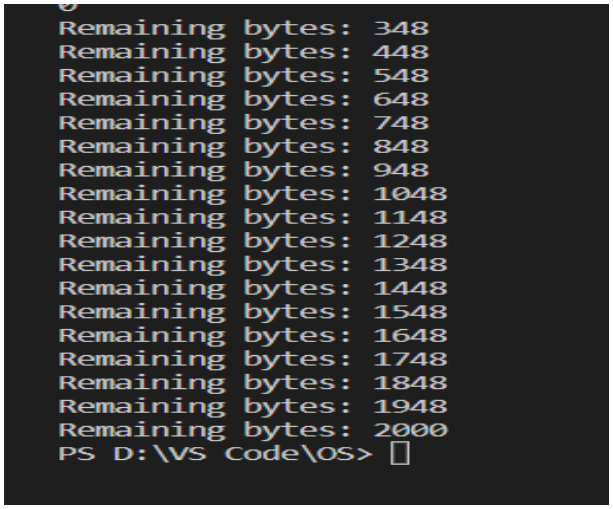
CODE:

```
#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:





OBSERVATION:

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

Code:

```
#include <stdio.h>
#include <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("Remaining bucket : %d \n", remaining);
printf("if you want to stop input press 0, else press 1");
scanf("%d", &k);
}
while (remaining < buckets)
{
    if (buckets - remaining > outlets)
    {
        remaining += outlets;
    }
    else
    remaing = buckets;
printf("Remaining byte : %d", remaining);
}
return 0;
}
```

Output

Enter Bucket size and outstream size

1000 200

Packets of 41 bytes are accepted
Remaining bytes 1000

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

Packet of 467 bytes are accepted
Remaining bytes 533

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

Packets of 334 bytes are accepted
Remaining bytes 599.