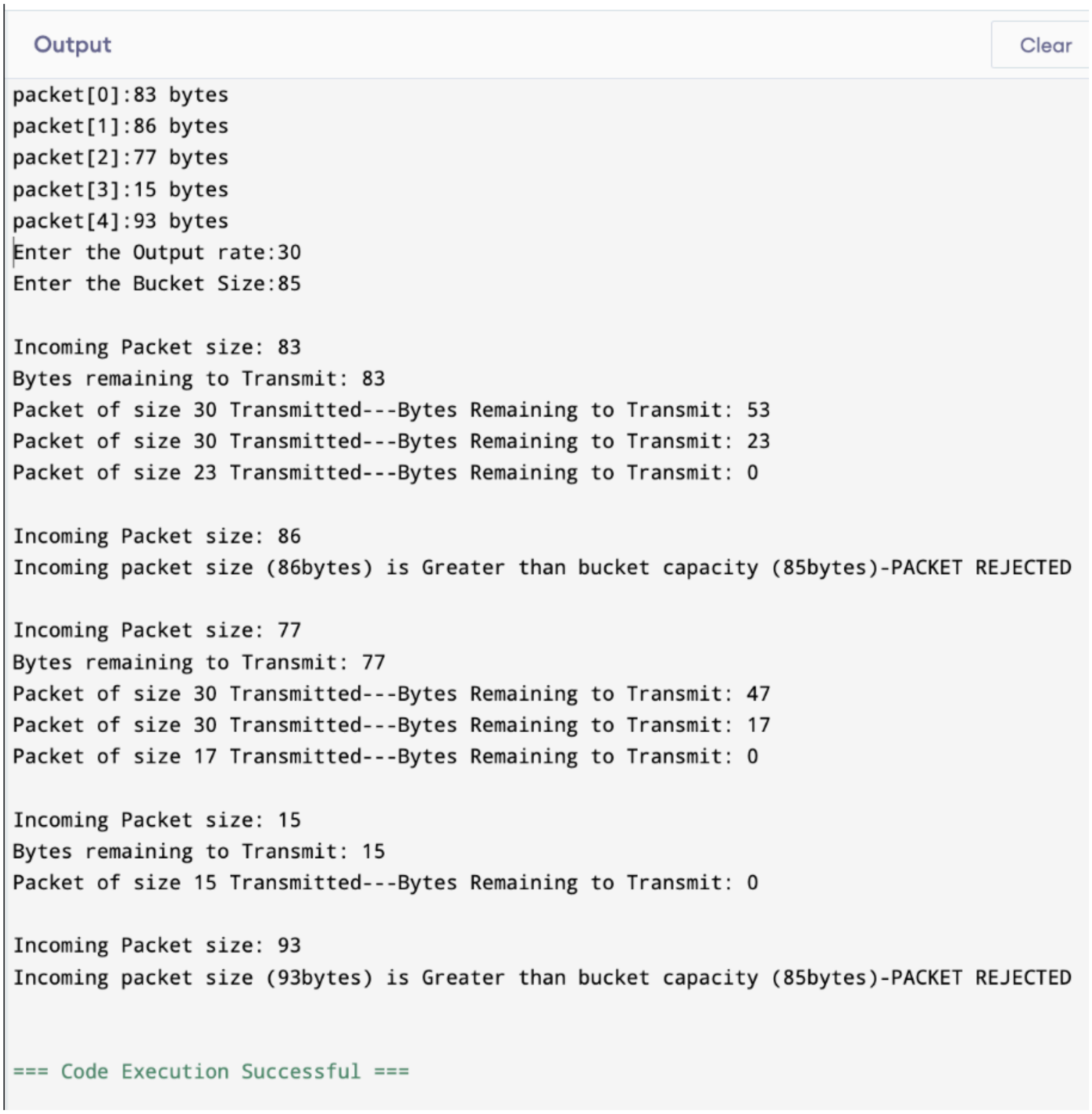

```
packet[0]:83 bytes
packet[1]:86 bytes
packet[2]:77 bytes
packet[3]:15 bytes
packet[4]:93 bytes
Enter the Output rate:30
Enter the Bucket Size:85

Incoming Packet size: 83
Bytes remaining to Transmit: 83
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 53
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 23
Packet of size 23 Transmitted---Bytes Remaining to Transmit: 0

Incoming Packet size: 86
Incoming packet size (86bytes) is Greater than bucket capacity (85bytes)-PACKET REJECTED

Incoming Packet size: 77
Bytes remaining to Transmit: 77
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 47
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 17
Packet of size 17 Transmitted---Bytes Remaining to Transmit: 0

Incoming Packet size: 15
Bytes remaining to Transmit: 15
Packet of size 15 Transmitted---Bytes Remaining to Transmit: 0

Incoming Packet size: 93
Incoming packet size (93bytes) is Greater than bucket capacity (85bytes)-PACKET REJECTED


=== Code Execution Successful ===
```

EXPERIMENT - 14

Leaky Bucket Program : Write a program for congestio
control using leaky bucket
algorithm.

```
#include <stdio.h>
 #include <stdlib.h>
#include <unistd.h>
#define NOF_PACKETS  5
int   rand (int a)
{
  int rn = (random() %10) % a
  return rn == 0 ? 1 : rn
 }

#include <stdlib.h>
  long int random (void);
 int main()
{ int packet_sz [NOF_PACKETS], i, clk, b_size, o_rate, p_sz_rm
     p_sz, p_time, op;
 ~~for~~ (i=0;  i<NOF_PACKETS ; i++)
       packet_sz[i]= random () %100;
   ~~for~~ for (i=0;  i<NOF_PACKETS , i++)
         printf("\npacket [%d] : %d bytes \t", i, packet_sz[i
  printf("\nEnter the output rate:");
  scanf ("%d", &o_rate);
  printf (" Enter the Bucket Size:");
  scanf ("%d", &b_size);
 for(i=0;  i<NOF_PACKETS ; i++)
   { if (packet_sz[i] + p_sz_rm) > b_size)
       if (packet_sz[i] > b_size)
         printf("\n\n Incoming packet size ("%d byte
           is greater than bucket capacity
  (%d bytes) - PACKET REJECTED", packet_sz[i], b_size);
     else
       printf("\n\n Bucket capacity exceeded - PACKETS
                REJECTED!!");
   else { p_sz_rm += packet_sz[i];
    printf ("\n\n Incoming Packet size:%d", packet_sz[
printf("\nBytes remaining to Transmitt : %d", p_sz_rem);
 p_time = random() * 10;
 printf ("\n Time left for ~~transmission~~: %d units",
        p_time );
  for (clk=10; clk <= p_time; clk+=10)
 while (p_sz_rem >0)
{
  sleep(1);
  if (p_sz_rem)
 {
   if (p_sz_rem <= o_rate)
     op= p_sz_rem , p_sz_rem= 0
else
     op = o_rate , p_sz_rem - o_rate;
  printf ("\n Packet of size %d Transmitted ", op);
printf ("----Bytes Remaining To Transmitt : %d", p_sz_rem
 }
 else
 {
   printf ( "\n No packets to transmitt !!");
 }
}}
}
```