Shaan Subbaiah B C

```
class LeakyBucket:
    def __init__(self, b_size, o_rate, packets):
        self.b_size = b_size
        self.o_rate = o_rate
        self.packets = packets

    def traffic_shaping(self):
        for i in range((len(self.packets))):
            packet_size = self.packets[i].
            print(f"Packet no: {i} Size: {packet_size}")
            if packet_size > b_size:
                print ("Bucket Overflow")
            else:
                while packet_size > o_size:
                    print(f"{o_rate} bytes sent")
                    packet_size -= o_rate

                if packet_size:
                    print(f"Last {packet_size} bytes sent

                print ("Bucket output successful")

bucket_size = int(input("Enter the bucket size: "))
o_rate = int(input("Enter the output rate: "))
packets = [int(x) for x in input("Enter the input packets").split()]
lb = LeakyBucket(b_size, o_rate, packet)
lb.traffic_shaping()
```