

3) Write a program to implement
Leaky bucket algorithm

```
import os
```

```
class Client:
```

```
    def __init__(self, rate=int, data=[]):
```

```
        self.rate = rate
```

```
        self.data = data
```

```
    def __str__(self):
```

```
        return str([str(self.rate),  
                    str(self.data)])
```

```
class Buffer:
```

```
    def __init__(self, buffer-size=int,  
                  buffer=[]):
```

```
        self.buffer-size = buffer-size
```

```
        self.buffer = buffer
```

```
    def checkstate(self):
```

```
        if len(self.buffer) == 0:
```

```
            return True
```

```
    def __str__(self):
```

```
        return str([str(self.buffer-size),  
                    str(self.buffer)]);
```

```
basestate = True
```

```
sec = 1
```

```
buffer = Buffer(int(input("Enter Buffer  
size")))
```

```
client = client(int(input("Enter acceptance  
rate")))
```

```
data-to-send = str
```

```
while basestate:
```

```
    data-to-send = input("Enter a  
    string send by client  
    server")
```

```
    count = 0
```

```
    if buffer.checkstate():
```

```
        for i in range(0, len(data-to-send)):
            if i < client.rate:
```

```
                client.data.append(data-to-  
                send[i])
```

```
            else:
```

```
                if count < buffer.buffer-size:
```

```
                    buffer.buffer.append  
                    (data-to-send[i])
```

```
                    count = len(buffer.buffer)
```

```
            else:
```

```
                print("Data loss" + data-to-  
                send[i])
```