In [4]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as sps
import math
import time
%matplotlib inline
```

In [5]:

```python
# Считаем данные
data = np.array([])
with open('servers.txt' , 'r' ) as f:
    for s in f:
        data = np.append(data, float(s))

l = data[0]
data = data[1:]
t0 = 1
t = 60
```

$E(N_t - N_s) = \lambda(t - s)$ (т.к. $N_t - N_s \sim Pois(\lambda(t - s))$)

$E(N_t|N_s) = E(N_t - N_s|N_s) + E(N_s|N_s) = E(N_t - N_s) + N_s = \lambda(t - s) + N_s$ (т.к. $N_t - N_s$ и N_s независимы)

In [6]:

```
Ns = 0
for s in range(0, t + 1, t0):
    # Будем для каждого момента времени s
    # Пересчитывать количество сломавшихся серверов
    while (Ns < len(data) and data[Ns] <= s):
        Ns += 1

    # Зная его посчитаем условное матожидание
    print('%d\t%f' % (s, l * (t - s) + Ns))
    time.sleep(t0)
```

0	18.300000
1	18.995000
2	18.690000
3	18.385000
4	18.080000
5	17.775000
6	17.470000
7	17.165000
8	16.860000
9	16.555000
10	16.250000
11	16.945000
12	17.640000
13	18.335000
14	18.030000
15	17.725000
16	17.420000
17	18.115000
18	17.810000
19	17.505000
20	17.200000
21	16.895000
22	16.590000
23	16.285000
24	17.980000
25	17.675000
26	18.370000
27	18.065000
28	18.760000
29	18.455000
30	18.150000
31	18.845000
32	19.540000
33	19.235000
34	18.930000
35	19.625000
36	19.320000
37	20.015000
38	19.710000
39	19.405000
40	21.100000
41	21.795000
42	22.490000
43	22.185000
44	21.880000
45	22.575000
46	22.270000
47	21.965000
48	21.660000
49	22.355000
50	22.050000
51	21.745000
52	23.440000
53	23.135000
54	22.830000
55	23.525000
56	23.220000

```
57        23.915000
58        23.610000
59        23.305000
60        25.000000
```

In [ ]:

In [ ]: