

In [26]:

```
import numpy as np
```

Получим формулу для $E(N_t|N_s)$ через $E(N_t - N_s|N_s)$. С одной стороны,

$$E(N_t - N_s|N_s) = E(N_t|N_s) - E(N_s|N_s) = E(N_t|N_s) - N_s \text{ (по линейности)}$$

С другой, так как $N_t - N_s$ и N_s независимы по условию

$$E(N_t - N_s|N_s) = E(N_t - N_s)$$

Значит,

$$E(N_t|N_s) = E(N_t - N_s) + N_s = \lambda \cdot (t - s) + N_s (\\ N_t - N_s \sim Pois(\lambda \cdot (t - s)))$$

In [27]:

```
# времена выхода из строя очередного сервера
times = np.loadtxt('6_1.txt')

lambda_ = times[0]
time = 60

# N_s - количество серверов, вышедших из строя к моменту времени S
N_s = np.zeros(61)

# счетчик сломавшихся серверов
counter = 1

# заполнение N_s
for i in range(61):
    while times[counter] < i and counter < (times.size - 2):
        counter = counter + 1
    N_s[i] = counter - 1

# вывод E(N_t / N_s)
for current_time in range(61):

    print current_time, "\t", (lambda_*(time - current_time) + N_s[current_time])
```

0	22.02
1	21.653
2	21.286
3	23.919
4	24.552
5	25.185
6	24.818
7	24.451
8	24.084
9	23.717
10	23.35
11	22.983
12	22.616
13	23.249
14	24.882
15	24.515
16	25.148
17	25.781
18	25.414
19	26.047
20	26.68
21	26.313
22	28.946
23	28.579
24	29.212
25	28.845
26	29.478
27	29.111
28	28.744
29	28.377
30	28.01
31	29.643
32	29.276
33	28.909
34	30.542
35	30.175
36	29.808
37	30.441
38	30.074
39	29.707
40	29.34
41	30.973
42	30.606
43	30.239
44	31.872
45	32.505
46	32.138
47	31.771
48	31.404
49	31.037
50	31.67
51	31.303
52	30.936
53	31.569
54	31.202
55	30.835
56	30.468

57	30.101
58	29.734
59	29.367
60	29.0

In [23]:

```
# выведем times.size, чтобы посмотреть, сколько у нас примерно должно получиться с  
ерверов  
print (times.size - 1) # (times.size - 1), потому что в первой строке на дана лям  
бда
```

31

Мы видим, что прогноз действительно очень похож на правду