

In [1]:

```
%pylab inline
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as sps
import time
```

Populating the interactive namespace from numpy and matplotlib

$(N_t - N_s) | N_s$  - независимы

$N_t - N_s \sim \text{Pois}(\lambda(t - s))$

$E(N_t - N_s) = \lambda(t - s)$

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

Извлекаем данные из файла:

In [2]:

```
strs = open("/Users/evgeniatveritinoval/Downloads/file2.txt").read().split()
times = np.zeros(len(strs)-1)
for i in range(len(strs) - 1):
    times[i] = float(strs[i + 1])
l = float(strs[0])
t = 60
```

Считаем по данным значение  $N_s$  каждую секунду до заданного времени

In [3]:

```
Ns = np.zeros(61)
for i in range(61):
    for t in times:
        if t <= i:
            Ns[i] += 1
```

Выводим посчитанные по формуле, выведенной выше,  $E(N_t | N_s)$ . То есть предсказания количества серверов, вышедших из строя к данному моменту.

In [4]:

```
for s in range(61):  
    print s, (1 * (t - s) + Ns[s])  
    time.sleep(1)
```

0 14.11176  
1 13.85176  
2 13.59176  
3 13.33176  
4 13.07176  
5 12.81176  
6 12.55176  
7 13.29176  
8 13.03176  
9 12.77176  
10 12.51176  
11 12.25176  
12 11.99176  
13 12.73176  
14 13.47176  
15 15.21176  
16 15.95176  
17 15.69176  
18 15.43176  
19 15.17176  
20 14.91176  
21 14.65176  
22 14.39176  
23 14.13176  
24 14.87176  
25 14.61176  
26 14.35176  
27 14.09176  
28 15.83176  
29 15.57176  
30 15.31176  
31 17.05176  
32 16.79176  
33 16.53176  
34 16.27176  
35 16.01176  
36 16.75176  
37 16.49176  
38 16.23176  
39 15.97176  
40 15.71176  
41 15.45176  
42 15.19176  
43 15.93176  
44 15.67176  
45 15.41176  
46 16.15176  
47 15.89176  
48 15.63176  
49 15.37176  
50 15.11176  
51 14.85176  
52 15.59176  
53 15.33176  
54 15.07176  
55 15.81176  
56 15.55176

57 15.29176  
58 15.03176  
59 14.77176  
60 14.51176