

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as sps
import math
import scipy
from collections import Counter
import copy
%matplotlib inline
```

```
In [10]: def build_time_plot(t1):
    n = np.arange(len(t1)) + 1
    plt.figure(figsize=(15,7))
    plt.grid(True)
    plt.title('time(x)')
    plt.xlabel('num of threads')
    plt.ylabel('seconds')
    plt.plot(n,t1,'-o')
    plt.show()

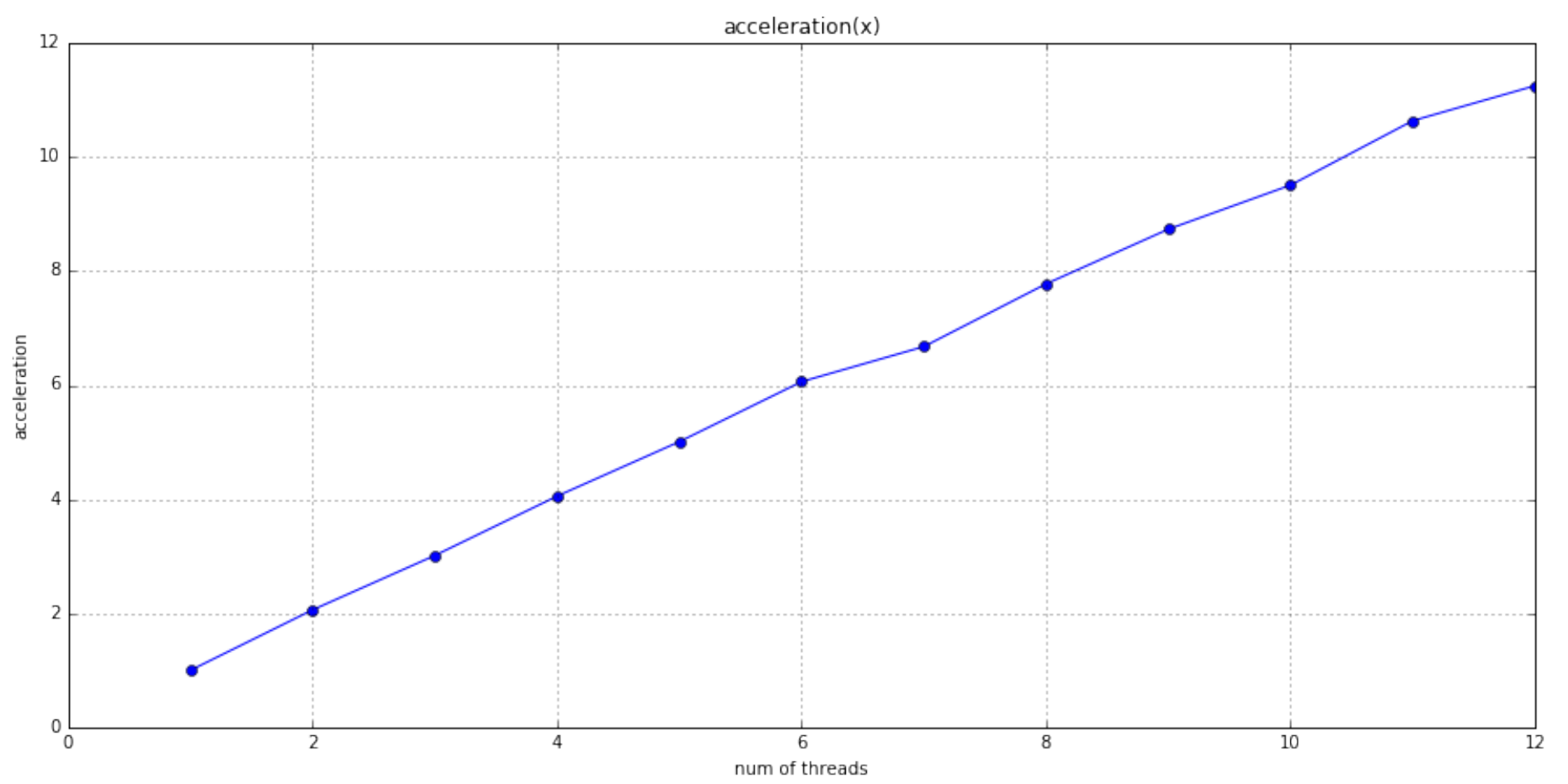
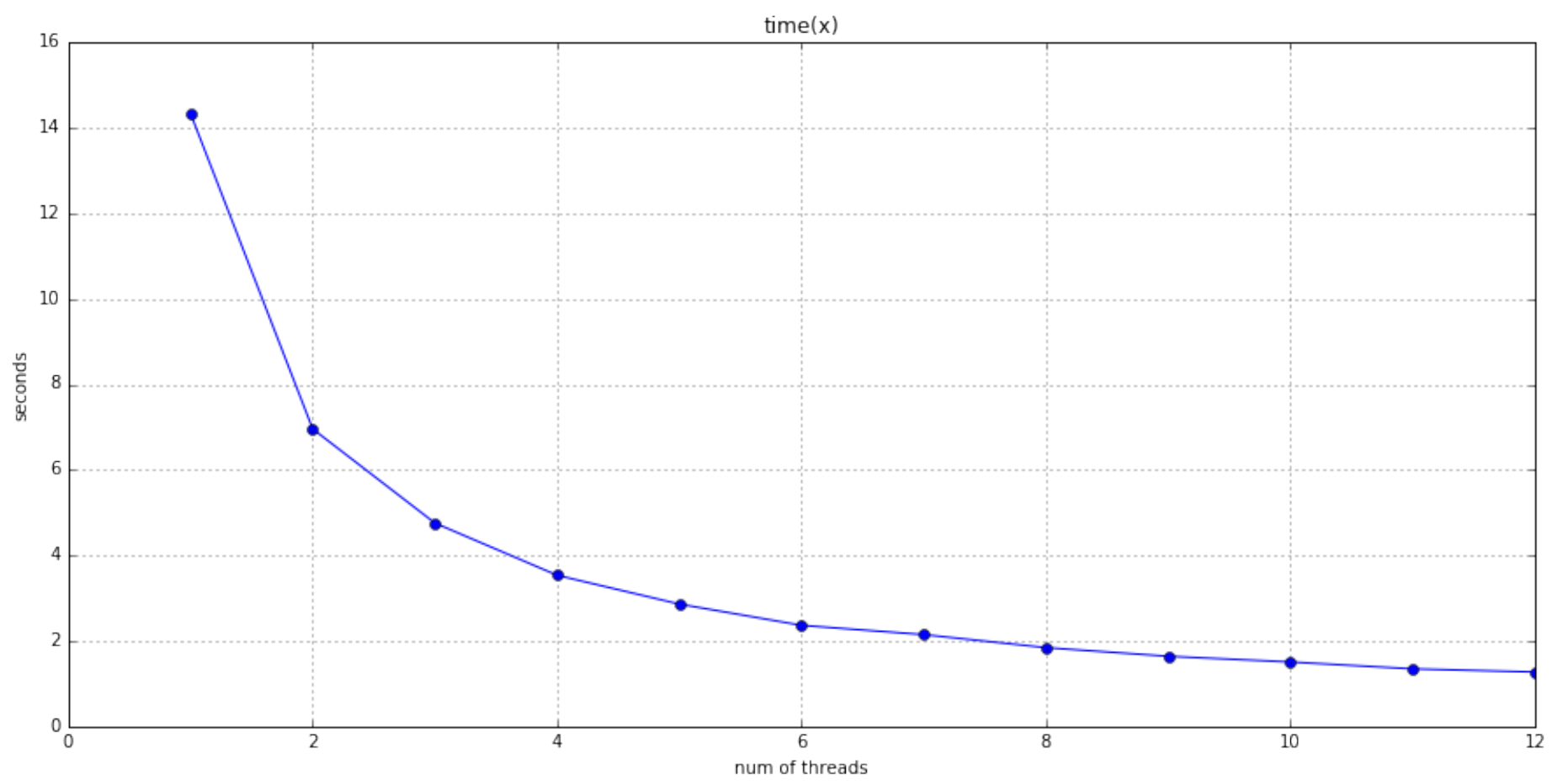
def build_acceleration_plot(t1) :
    n = np.arange(len(t1)) + 1
    aclr = [t1[0]/t for t in t1]

    plt.figure(figsize=(15,7))
    plt.grid(True)
    plt.title('acceleration(x)')
    plt.xlabel('num of threads')
    plt.ylabel('acceleration')
    plt.plot(n,aclr,'-o')
    plt.show()
```

1

Без семафоров, возвращение через join

```
In [13]: t1 = [14.337314 ,6.957273 ,4.756307 ,3.535787 ,2.861834 ,2.365084,\
    2.147230 ,1.843590 ,1.641226 ,1.507750 ,1.347671 ,1.274207]
build_time_plot(t1)
build_acceleration_plot(t1)
```



2

Общая переменная

```
In [14]: t2 = [14.132047, 7.079802, 4.710376, 3.627829, 2.841821,\n              2.364726, 2.099244, 1.858679, 1.636391, 1.494669, 1.477940, 1.242244]\n          n = np.arange(len(t1)) + 1\n\n          build_time_plot(t2)\n          build_acceleration_plot(t2)
```

