Task 1
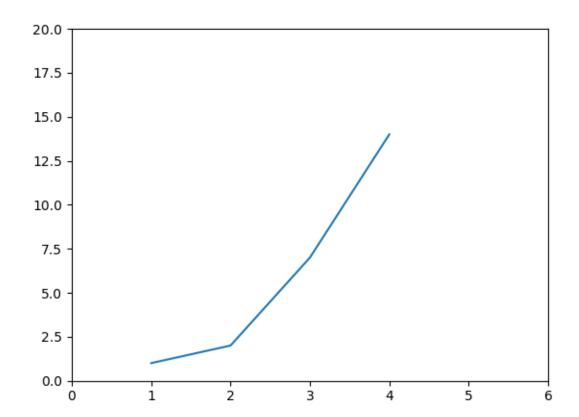


```
C:\Windows\system32\cmd.exe

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Qiang>conda info
Current conda install:

               platform : win-64
          conda version : 4.3.24
       conda is private : False
      conda-env version : 4.3.24
    conda-build version : not installed
         python version : 3.6.1.final.0
       requests version : 2.14.2
       root environment : C:\Users\Qiang\Anaconda3   (writable)
    default environment : C:\Users\Qiang\Anaconda3
       envs directories : C:\Users\Qiang\Anaconda3\envs
                          C:\Users\Qiang\AppData\Local\conda\conda\envs
                          C:\Users\Qiang\.conda\envs
          package cache : C:\Users\Qiang\Anaconda3\pkgs
                          C:\Users\Qiang\AppData\Local\conda\conda\pkgs
           channel URLs : https://conda.anaconda.org/anaconda-fusion/win-64
                          https://conda.anaconda.org/anaconda-fusion/noarch
                          https://repo.continuum.io/pkgs/free/win-64
                          https://repo.continuum.io/pkgs/free/noarch
                          https://repo.continuum.io/pkgs/r/win-64
                          https://repo.continuum.io/pkgs/r/noarch
                          https://repo.continuum.io/pkgs/pro/win-64
                          https://repo.continuum.io/pkgs/pro/noarch
                          https://repo.continuum.io/pkgs/msys2/win-64
                          https://repo.continuum.io/pkgs/msys2/noarch
            config file : C:\Users\Qiang\.condarc
             netrc file : None
           offline mode : False
             user-agent : conda/4.3.24 requests/2.14.2 CPython/3.6.1 Windows/7 W
indows/6.1.7601
          administrator : False

C:\Users\Qiang>
```

Task 2


```
   IPython: C:Anaconda3/Scripts                                        _  □  x
?             -> Introduction and overview of IPython's features.
quickref  -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]: import numpy as np

In [2]: import scipy.linalg

In [3]: a = np.array([[1,2,3,4,5],[6,7,8,9,10]])

In [4]: print(a.ndim)
2

In [5]: print(a.size)
10

In [6]: print(a.shape)
(2, 5)

In [7]: print(a.shape[1])
5

In [8]: a[-1]
Out[8]: array([ 6,  7,  8,  9, 10])

In [9]: print(a[-1])
[ 6  7  8  9 10]

In [10]: print(a[1,1])
7

In [11]: print(a[1][1:3])
[7 8]
```
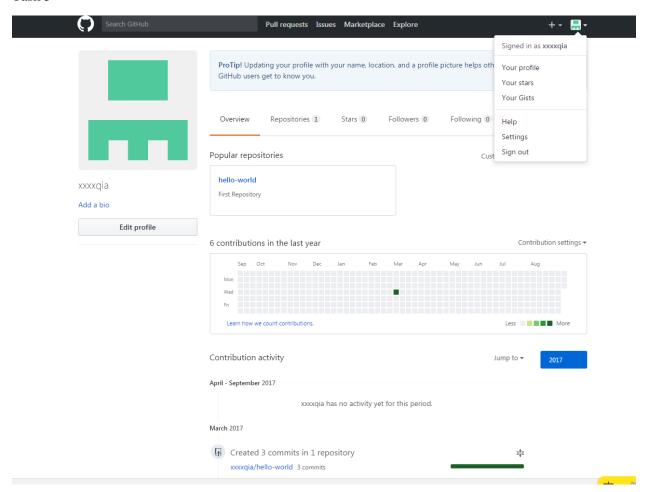
```
In [14]: print(a.T)
[[ 1  6]
 [ 2  7]
 [ 3  8]
 [ 4  9]
 [ 5 10]]

In [15]: print(a.conj().T)
[[ 1  6]
 [ 2  7]
 [ 3  8]
 [ 4  9]
 [ 5 10]]

In [16]: print(a*3)
[[ 3  6  9 12 15]
 [18 21 24 27 30]]

In [17]: print(a/3)
[[ 0.33333333  0.66666667  1.          1.33333333  1.66666667]
 [ 2.          2.33333333  2.66666667  3.          3.33333333]]

In [18]: print(a**3)
[[   1    8   27   64  125]
 [ 216  343  512  729 1000]]

In [19]: print(a>3)
[[False False False  True  True]
 [ True  True  True  True  True]]
```

Task 4

```python
import matplotlib.pyplot as plt
plt.plot([1,2,3,4],[1,2,7,14])
plt.axis([0,6,0,20])
plt.show()
```

Task 5

Task 6

https://github.com/xxxxqia/COMP576