

**宁波理工学院**

信号与系统实验报告

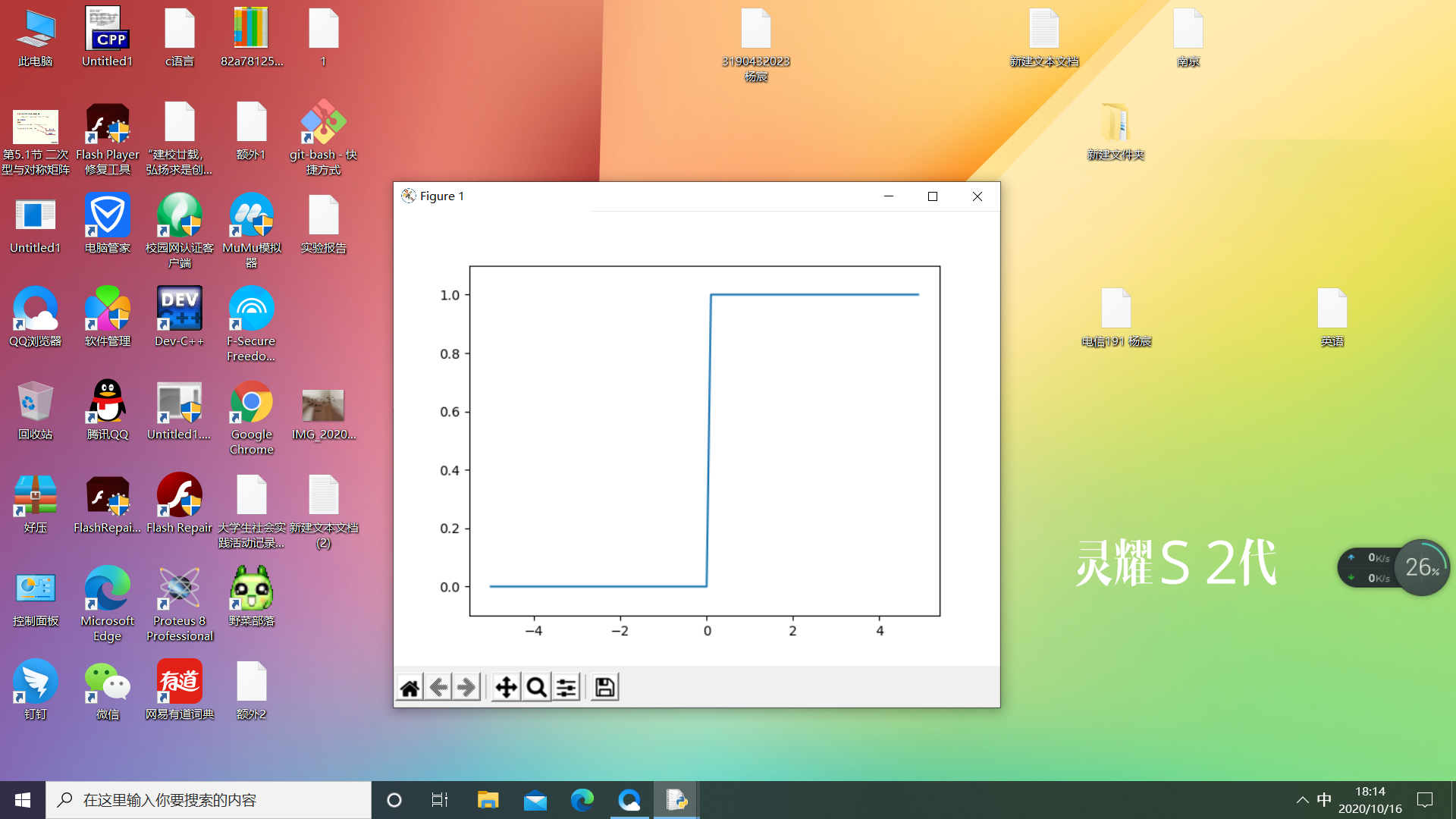


姓 名： 杨宸

学 号：  3190432023

专业班级： 电信191

学 院： 信息科学与工程学院

****

import numpy as np

import matplotlib.pylab as plt

def step\_function1(x=''):

'''

'''

if x >0:

return 1

else:

return 0

def step\_function2(x):

'''

'''

y = x > 0

return y.astype(np.int)

m = 1

print(step\_function1(m))

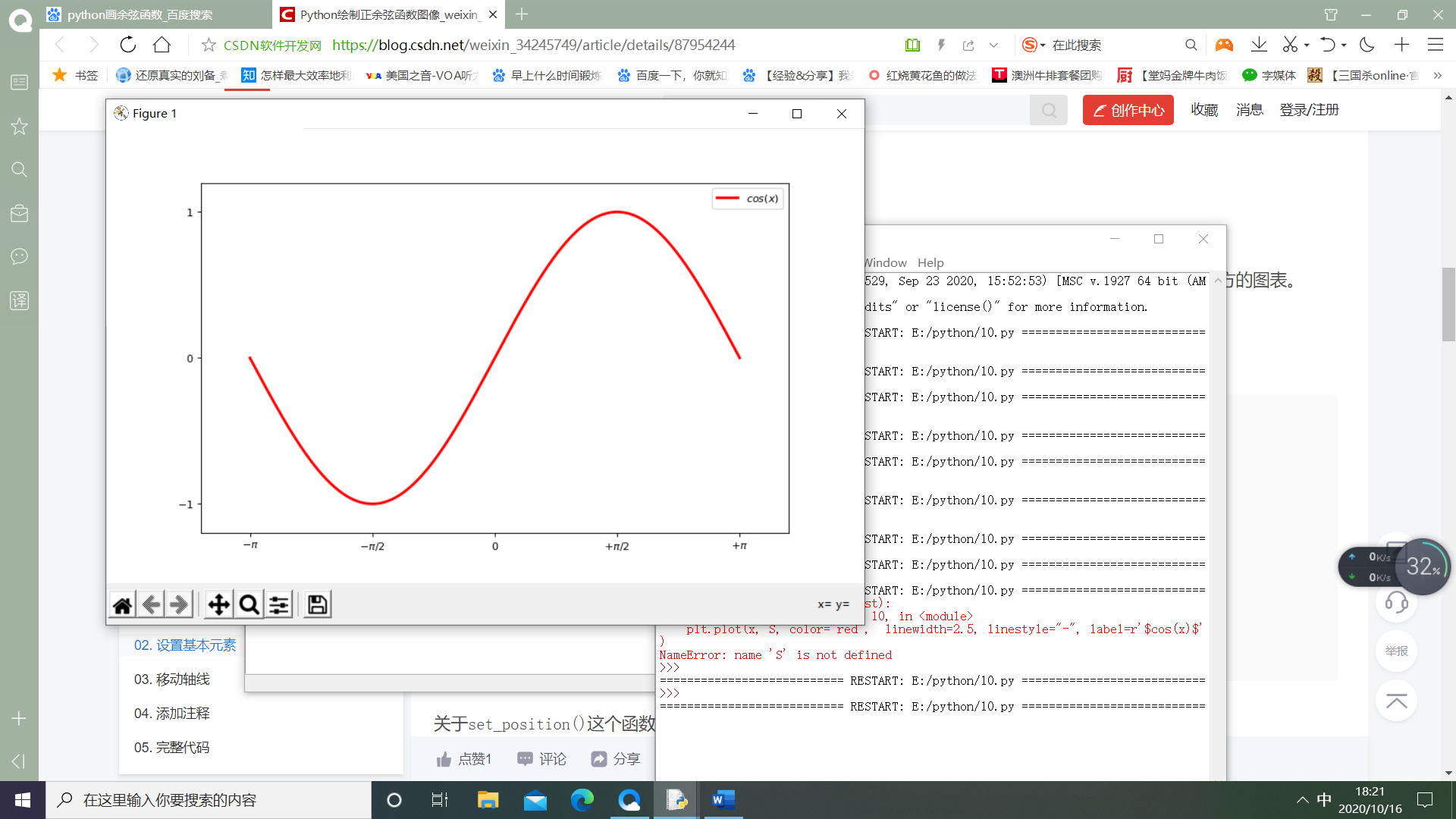
x = np.arange(-5.0,5.0,0.1)

y = step\_function2(x)

plt.plot(x,y)

plt.ylim(-0.1,1.1)

plt.show()

****

import numpy as np

from matplotlib import pyplot as plt

plt.figure(figsize=(10,6), dpi=80)

x = np.linspace(-np.pi, np.pi, 256,endpoint=True)

C,S = np.cos(x), np.sin(x)

plt.plot(x, S, color="red", linewidth=2.5, linestyle="-", label=r'$cos(x)$')

plt.xlim(x.min()\*1.2, x.max()\*1.2)

plt.ylim(C.min()\*1.2, C.max()\*1.2)

plt.xticks([-np.pi, -np.pi/2, 0, np.pi/2, np.pi],

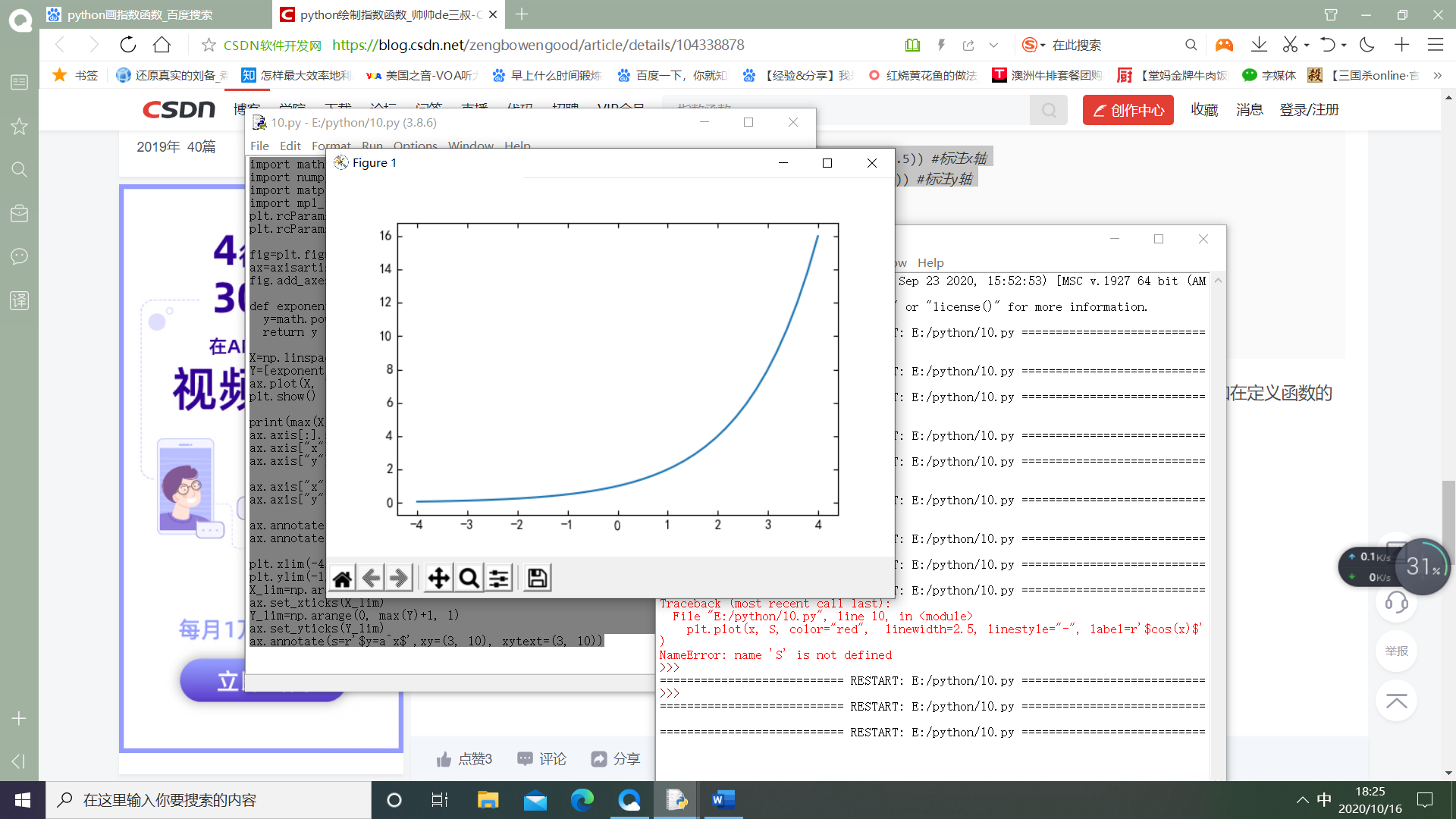
[r'$-\pi$', r'$-\pi/2$', r'$0$', r'$+\pi/2$', r'$+\pi$'])

plt.yticks([-1,0,1],

[r'$-1$', r'$0$', r'$1$'])

plt.legend()

plt.show()



import math

import numpy as np

import matplotlib.pyplot as plt

import mpl\_toolkits.axisartist as axisartist

plt.rcParams['font.sans-serif']=['SimHei']

plt.rcParams['axes.unicode\_minus']=False

fig=plt.figure(figsize=(6,4))

ax=axisartist.Subplot(fig,111)

fig.add\_axes(ax)

def exponential\_func(x, a=2):

y=math.pow(a, x)

return y

X=np.linspace(-4, 4, 40)

Y=[exponential\_func(x) for x in X]

ax.plot(X, Y)

plt.show()

print(max(X), max(Y))

ax.axis[:].set\_visible(False)

ax.axis["x"]=ax.new\_floating\_axis(0, 0, axis\_direction="bottom")

ax.axis["y"]=ax.new\_floating\_axis(1, 0, axis\_direction="bottom")

ax.axis["x"].set\_axisline\_style("-|>", size=1.0)

ax.axis["y"].set\_axisline\_style("-|>", size=1.0)

ax.annotate(s='x', xy=(max(X), 0), xytext=(max(X)+0.5, 0.5))

ax.annotate(s='y', xy=(0, 1.0), xytext=(-0.5, max(Y)+0.5))

plt.xlim(-4, 5)

plt.ylim(-1, 17)

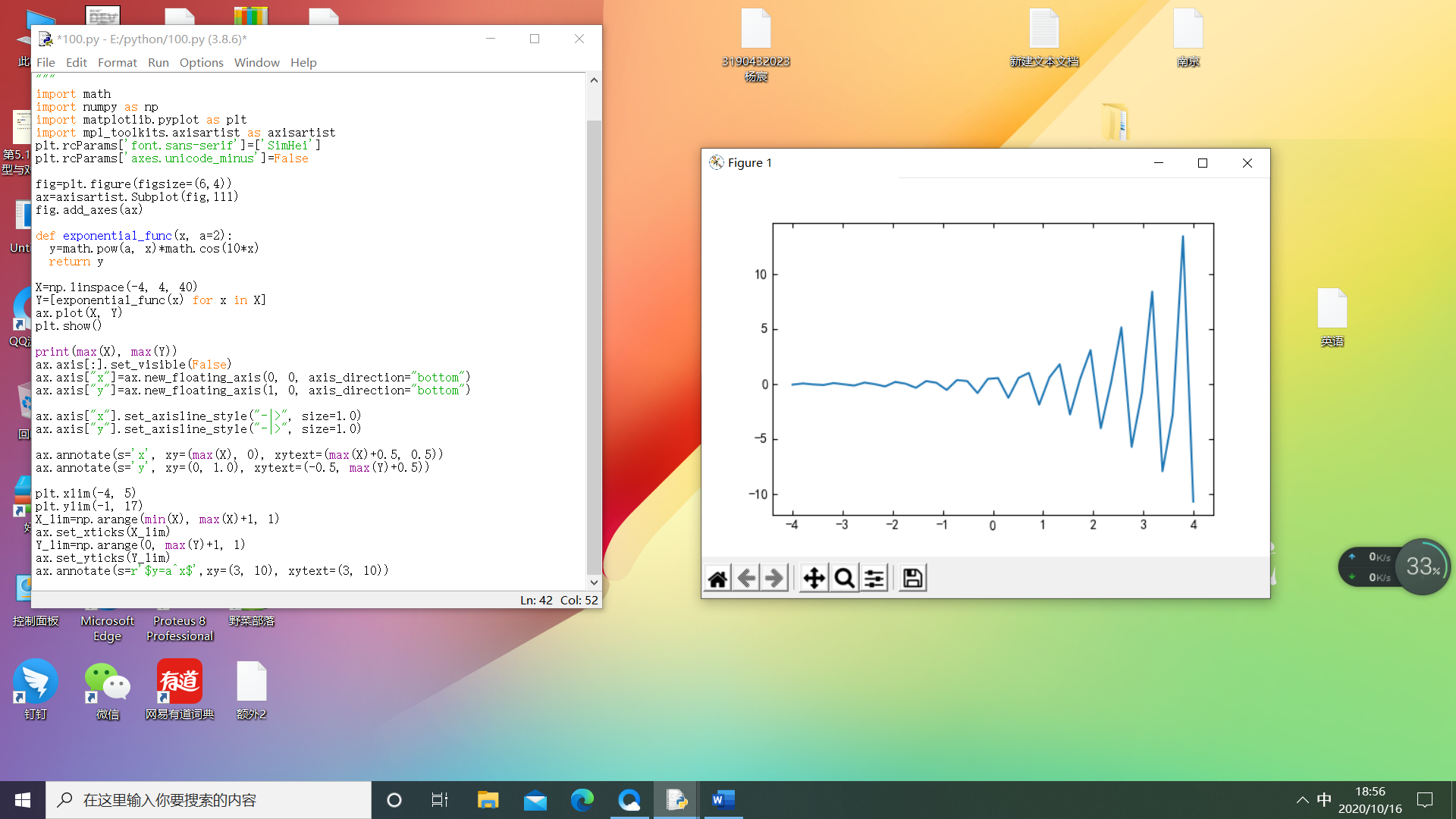
X\_lim=np.arange(min(X), max(X)+1, 1)

ax.set\_xticks(X\_lim)

Y\_lim=np.arange(0, max(Y)+1, 1)

ax.set\_yticks(Y\_lim)

ax.annotate(s=r'$y=a^x$',xy=(3, 10), xytext=(3, 10))



# -\*- coding: utf-8 -\*-

"""

"""

import math

import numpy as np

import matplotlib.pyplot as plt

import mpl\_toolkits.axisartist as axisartist

plt.rcParams['font.sans-serif']=['SimHei']

plt.rcParams['axes.unicode\_minus']=False

fig=plt.figure(figsize=(6,4))

ax=axisartist.Subplot(fig,111)

fig.add\_axes(ax)

def exponential\_func(x, a=2):

y=math.pow(a, x)\*math.cos(10\*x)

return y

X=np.linspace(-4, 4, 40)

Y=[exponential\_func(x) for x in X]

ax.plot(X, Y)

plt.show()

print(max(X), max(Y))

ax.axis[:].set\_visible(False)

ax.axis["x"]=ax.new\_floating\_axis(0, 0, axis\_direction="bottom")

ax.axis["y"]=ax.new\_floating\_axis(1, 0, axis\_direction="bottom")

ax.axis["x"].set\_axisline\_style("-|>", size=1.0)

ax.axis["y"].set\_axisline\_style("-|>", size=1.0)

ax.annotate(s='x', xy=(max(X), 0), xytext=(max(X)+0.5, 0.5))

ax.annotate(s='y', xy=(0, 1.0), xytext=(-0.5, max(Y)+0.5))

plt.xlim(-4, 5)

plt.ylim(-1, 17)

X\_lim=np.arange(min(X), max(X)+1, 1)

ax.set\_xticks(X\_lim)

Y\_lim=np.arange(0, max(Y)+1, 1)

ax.set\_yticks(Y\_lim)

ax.annotate(s=r'$y=a^x$',xy=(3, 10), xytext=(3, 10))