Python第四次作业

19338035黄海俊

源代码修正及作业如下：

#PythonWorkT3V1

# numpy是一个重要的Python数值程序库，包含了大量的数组运算函数

import numpy as np

# scipy是一个重要的Python数值程序库，包含了优化计算、符号计算等各类函数

from scipy.interpolate import interp1d

# matplotlib是一个重要的Python绘图数据库

from matplotlib import pyplot as plt

# 收入的阈值，有7个档次

thresholds\_revenue = np.array([0, 36000, 144000, 300000, 420000, 660000, 960000])

# 针对收入的7个档次，有7种税率

thresholds\_taxrate = np.array([0.03, 0.10, 0.20, 0.25, 0.30, 0.35, 0.45])

# 在各个档次之间，个人所得税随税率呈现出线性增加的特征；相应的，所得税计算的重点，在于确定各个档次阈值的缴税

revenues = [] # 税前收入序列

revenues\_minus\_taxes = [] # 税后收入序列

taxes = [] # 缴税金额序列

速算扣除数 = [] # 速算扣除数序列

# 第零档：都从零开始

revenues.append(0.00)

revenues\_minus\_taxes.append(0.00)

taxes.append(0.00)

# 第一至六档的计算

for i in range(1,

len(thresholds\_revenue)):

# 税前收入阈值

revenues.append(thresholds\_revenue[i])

# 所得税增值

tax\_delta = (revenues[i] - revenues[i - 1]) \* thresholds\_taxrate[i - 1]

# 所得税总值

tax = taxes[i - 1] + tax\_delta

# 所得税序列

taxes.append(tax)

# 税后收入

revenues\_minus\_taxes.append(revenues[-1] - taxes[-1])

# 速算扣除数

速算扣除数.append(revenues[-1] \* thresholds\_taxrate[i - 1] - taxes[-1])

print(速算扣除数)

# 第七档的计算

revenues.append(thresholds\_revenue[-1] + 20000)

taxes.append(taxes[-1] + (revenues[-1] - revenues[-2]) \* thresholds\_taxrate[-1])

revenues\_minus\_taxes.append(revenues[-1] - taxes[-1])

# 税前与税后收入序列

revenues = np.array(revenues)

revenues\_minus\_taxes = np.array(revenues\_minus\_taxes)

# 构建线性插值函数

cal\_revenue\_after\_tax = interp1d(x=revenues, y=revenues\_minus\_taxes,

kind='linear')

# 计算，并打印计算结果

'''

问题1：

根据税前收入、所得税等数据，如何计算得到“速算扣除数”？

速算扣除数 = 本金 \* 对应档税率 - 实际税额

如何调整for循环中的代码，输出各个档次收入的速算扣除数？

如上所示

'''

for i, revenue in enumerate(revenues):

revenue\_after\_tax = cal\_revenue\_after\_tax(revenue)

tax = revenue - revenue\_after\_tax

print('档次 {}:\n 税前收入: {}\n 税后收入: {}\n 所得税额: {}\n'.format(i,

revenue,

revenue\_after\_tax,

tax,

))

# 绘图展示

revenues\_new = 10000 \* np.arange(98 + 1)

'''

问题2：

将上述语句改为“revenues\_new = 10000 \* np.arange(100 + 1)”，程序将报错

通读运行错误的提示信息，请问问题出在什么地方？

参数revenues的上限为98W，改成100后将超出参数的上限。

更改interp1d的设置，可以避免报错。请问如何更改？

revenues.append(thresholds\_revenue[-1] + 20000)中的20000改为40000

'''

revenues\_new\_minus\_taxes = cal\_revenue\_after\_tax(revenues\_new)

taxes\_new = revenues\_new - revenues\_new\_minus\_taxes

# 绘图区域尺寸设置

f = plt.figure(figsize=(10, 5))

# 左侧图形：税后收入与税前收入的关系

ax1 = f.add\_subplot(1, 2, 1)

ax1.plot(revenues\_new,

revenues\_new\_minus\_taxes, '-',

linewidth=2)

ax1.set\_xlabel('Revenue')

ax1.set\_ylabel('Revenue after tax')

# 右侧图形：所得税与税前收入的关系

ax2 = f.add\_subplot(1, 2, 2)

ax2.plot(revenues\_new,

taxes\_new, '-',

linewidth=2)

ax2.set\_xlabel('Revenue')

ax2.set\_ylabel('Tax')

'''

问题3：

如何编写程序命令，自动保存上述图形？

输入指令plt.savefig('./test2.jpg')

把绘图区域尺寸调整为（4,2），会有什么效果？为什么会出现这种效果？

图形右侧界限来回收缩，区域宽度尺寸不大于两个图形本身的宽度。

'''

plt.show()