Python实验报告15

班级：17应用统计学1班

姓名：王贵珍

学号：117060400124

指导老师：林卫中

实验题目：练习8.1 乒乓球比赛规则如下：每人发两个球，轮换发球权，胜方得分（无论是否为发球方），先得11分者胜。

实验代码：

from random import random

def printIntro():

print("这个程序模拟两个选手A和B的某种竞技比赛")

print("程序运行需要A和B的能力值（以0到1之间的小数表示）")

def getInputs():

a = eval(input("请输入选手A的能力值(0-1): "))

b = eval(input("请输入选手B的能力值(0-1): "))

n = eval(input("模拟比赛的场次: "))

return a, b, n

def simNGames(n, probA, probB):

winsA, winsB = 0, 0

for i in range(n):

scoreA, scoreB = simOneGame(probA, probB)

if scoreA > scoreB:

winsA += 1

else:

winsB += 1

return winsA, winsB

def gameOver(a,b):

return a==11 or b==11

def simOneGame(probA, probB):

scoreA, scoreB = 0, 0

serving = 0#0:表示A发球 1：表示B发球

i = 0

while not gameOver(scoreA, scoreB):

if serving == 0:

if random() < probA:

scoreA += 1

else:

scoreB += 1

else:

if random() < probB:

scoreB += 1

else:

scoreA += 1

i += 1

if i%2 ==0:

serving = (serving + 1)%2

return scoreA, scoreB

def printSummary(winsA, winsB):

n = winsA + winsB

print("竞技分析开始，共模拟{}场比赛".format(n))

print("选手A获胜{}场比赛，占比{:0.1%}".format(winsA, winsA/n))

print("选手B获胜{}场比赛，占比{:0.1%}".format(winsB, winsB/n))

def main():

printIntro()

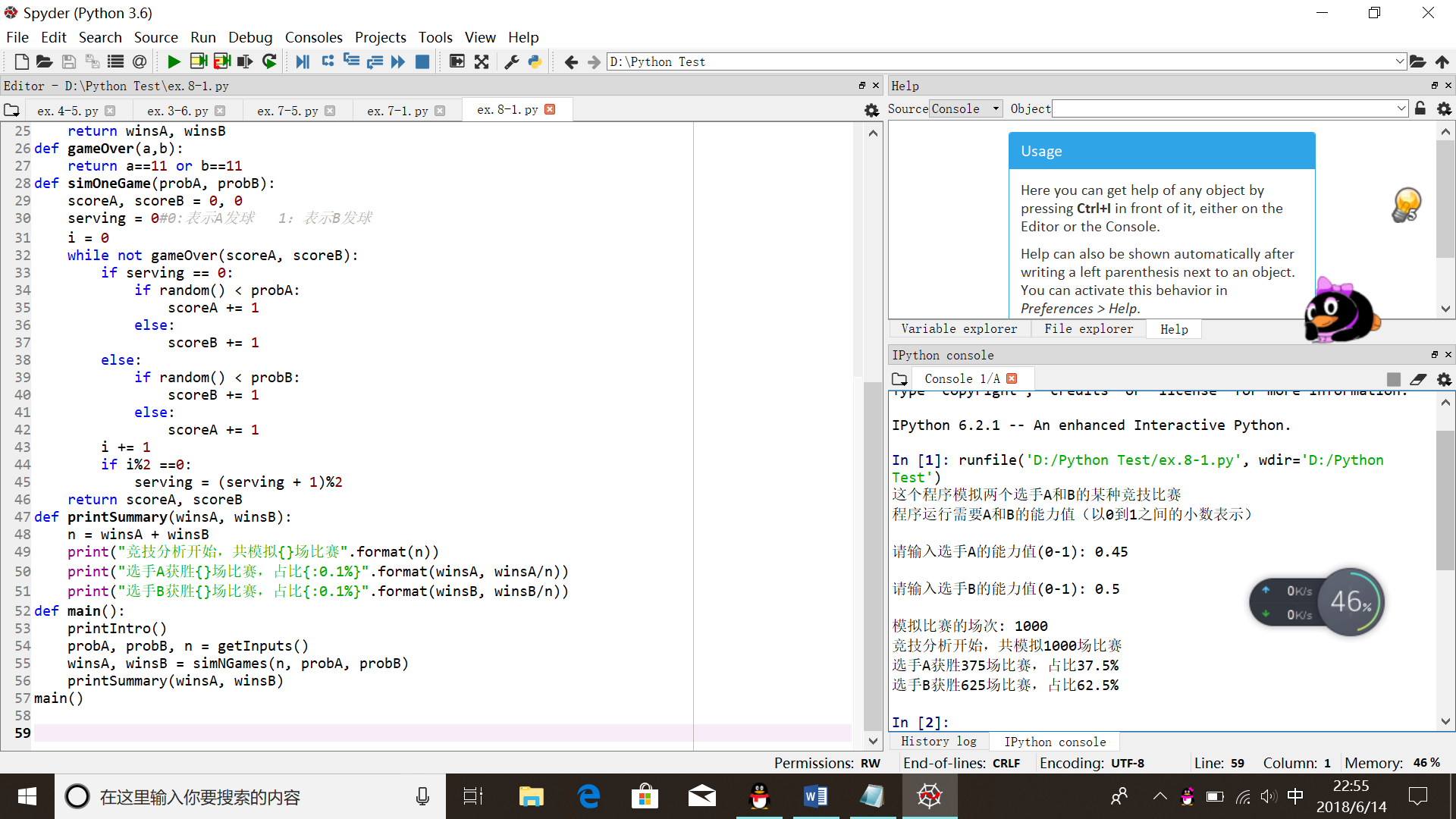
probA, probB, n = getInputs()

winsA, winsB = simNGames(n, probA, probB)

printSummary(winsA, winsB)

main()

实验结果：



扩展：先得11分为胜，10平后，先多得2分为胜方

from random import random

def printIntro():

print("这个程序模拟两个选手A和B的某种竞技比赛")

print("程序运行需要A和B的能力值（以0到1之间的小数表示）")

def getInputs():

a = eval(input("请输入选手A的能力值(0-1): "))

b = eval(input("请输入选手B的能力值(0-1): "))

n = eval(input("模拟比赛的场次: "))

return a, b, n

def simNGames(n, probA, probB):

winsA, winsB = 0, 0

for i in range(n):

scoreA, scoreB = simOneGame(probA, probB)

if scoreA > scoreB:

winsA += 1

else:

winsB += 1

return winsA, winsB

def gameOver(a,b):

if a<b:

a,b = b,a

if a ==11 and b<10:

return True

if b>=10 and a-b==2:

return True

return False

‘’‘

if (a < 10 and b==11) or (b < 10 and a==11):

return True

if (a>=10 and b-a == 2) or (b >=10 and a-b == 2):

return True

return False

'''

def simOneGame(probA, probB):

scoreA, scoreB = 0, 0

serving = 0#0:表示A发球 1：表示B发球

i = 0

while not gameOver(scoreA, scoreB):

if serving == 0:

if random() < probA:

scoreA += 1

else:

scoreB += 1

else:

if random() < probB:

scoreB += 1

else:

scoreA += 1

i += 1

if i%2 ==0:

serving = (serving + 1)%2

return scoreA, scoreB

def printSummary(winsA, winsB):

n = winsA + winsB

print("竞技分析开始，共模拟{}场比赛".format(n))

print("选手A获胜{}场比赛，占比{:0.1%}".format(winsA, winsA/n))

print("选手B获胜{}场比赛，占比{:0.1%}".format(winsB, winsB/n))

def main():

printIntro()

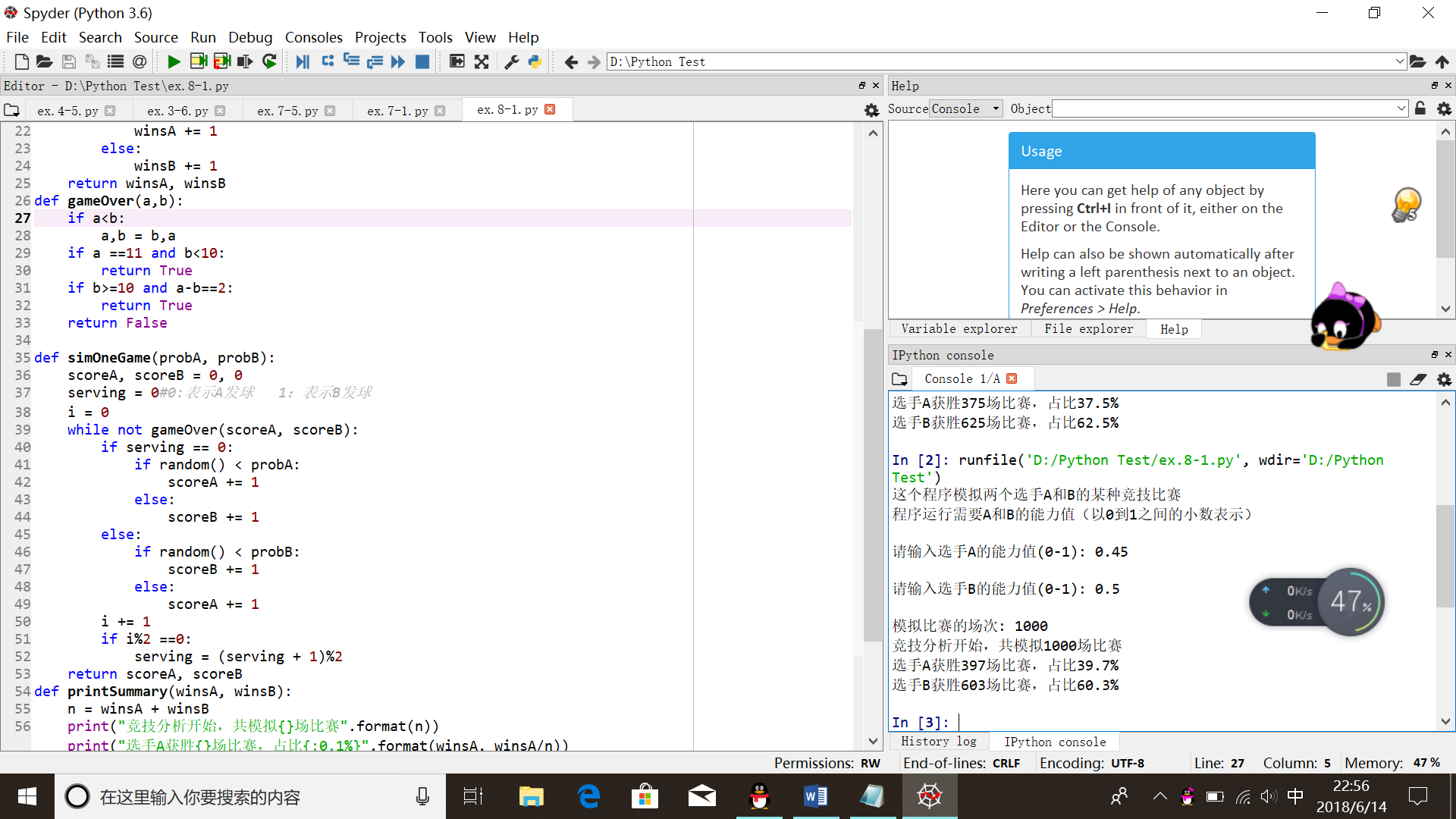
probA, probB, n = getInputs()

winsA, winsB = simNGames(n, probA, probB)

printSummary(winsA, winsB)

main()

实验结果：



实验题目：8.2 规则如下：甲乙两队比赛，考虑双方的投篮得分和篮板球能力水平。家对现货发球权，如果甲投篮中，则得1分，发球权交给乙方。如果甲投篮不中，双方抢篮板球，抢得篮板球一方获得球权，每队进攻限时24秒，若超过24秒则交换球权，每局比赛限时12分钟。

实验代码：

from random import random

from random import randint

def printIntro():

print("这个程序模拟两个选手A和B的某种竞技比赛")

print("程序运行需要A和B的能力值（以0到1之间的小数表示）")

def getInputs():

g1 = eval(input("请输入球队A的投篮能力值(0-1): "))

b1 = eval(input("请输入球队A的篮板能力值(0-1): "))

g2 = eval(input("请输入球队B的投篮能力值(0-1): "))

b2 = eval(input("请输入球队B的篮板能力值(0-1): "))

n = eval(input("模拟比赛的场次: "))

return g1, b1, g2, b2, n

def simNGames(n, goleA, boardA, goleB, boardB):

winsA, winsB = 0, 0

for i in range(n):

scoreA, scoreB = simOneGame(goleA, boardA, goleB, boardB)

if scoreA > scoreB:

winsA += 1

else:

winsB += 1

return winsA, winsB

def gameOver(t):

return t >= 12\*60

def simOneGame(goleA, boardA, goleB, boardB):

scoreA, scoreB = 0, 0

serving = 0 #0: 代表A队发球，1：代表B队发球

totalTime = 0

while not gameOver(totalTime):

t = randint(1, 24)

totalTime += t

if t == 24:

serving = (serving + 1)%2

else:

if serving == 0:

if random() < goleA:

scoreA += 1

serving = 1

else:

if random() < boardA:

serving=0

else:

serving = 1

else:

if random() < goleB:

scoreB += 1

serving = 0

else:

if random() < boardB:

serving = 1

else:

serving=0

return scoreA, scoreB

def printSummary(winsA, winsB):

n = winsA + winsB

print("竞技分析开始，共模拟{}场比赛".format(n))

print("选手A获胜{}场比赛，占比{:0.1%}".format(winsA, winsA/n))

print("选手B获胜{}场比赛，占比{:0.1%}".format(winsB, winsB/n))

def main():

printIntro()

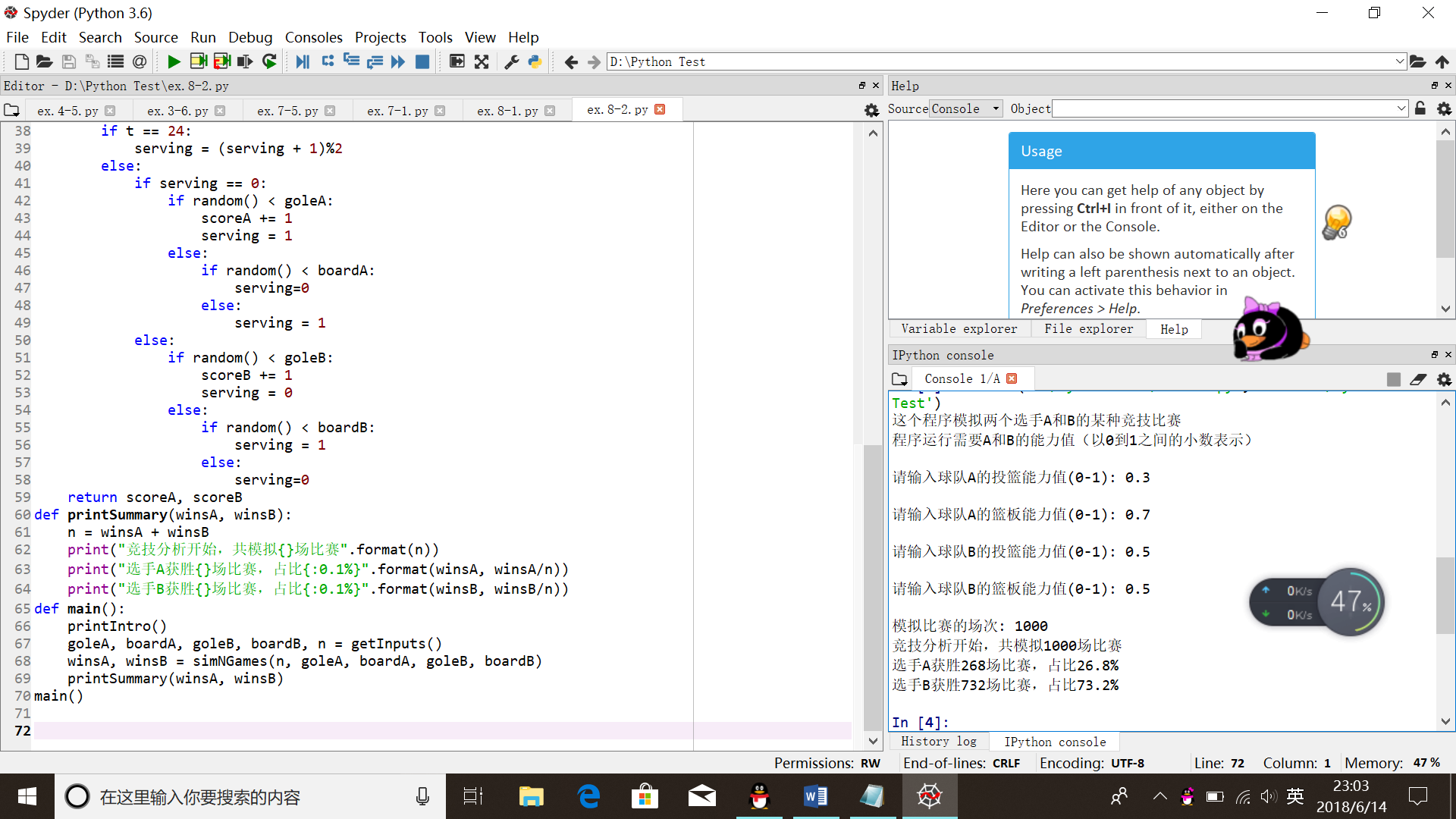
goleA, boardA, goleB, boardB, n = getInputs()

winsA, winsB = simNGames(n, goleA, boardA, goleB, boardB)

printSummary(winsA, winsB)

main()

实验结果：



实验题目：8.4 词云

实验代码：

import jieba

import matplotlib.pyplot as plt

from wordcloud import WordCloud, ImageColorGenerator

import numpy as np

import PIL.Image as Image

def calWordFrequence():

excludes = {} # {"将军","却说","丞相"}

txt = open("三国演义.txt", "r", encoding='utf-8').read()

words = jieba.lcut(txt)

counts = {}

for word in words:

if len(word) == 1:

continue

else:

counts[word] = counts.get(word, 0) + 1

for word in excludes:

del (counts[word])

return counts

def drawWordCloud(counts):

coloring = np.array(Image.open("E:/baidupic/9.png"))

wc = WordCloud(background\_color="white",

max\_words=2000,

mask=coloring,

max\_font\_size=60,

random\_state=42,

scale=2,

font\_path="c:/Windows/Fonts/SimHei.ttf")

wc.generate\_from\_frequencies(counts)

image\_colors = ImageColorGenerator(coloring)

plt.imshow(wc)

plt.axis("off")

plt.show()

def main():

counts = calWordFrequence()

drawWordCloud(counts)

main()