2022年10月－实验三

【实验名称】

Python程序控制结构练习

【实验目的】

1、编写程序，从键盘输入一个5位数字，判断这个数字是不是回文数（设n是一任意自然数，如果n的各位数字反向排列所得自然数与n相等，则n被称为回文数）。

2、编写程序，根据用户输入的一个18位合法身份证号，输出用户的出生年月日、年龄和性别（第7-10位为出生年份，第11-12位为出生月份，第13-14位代表出生日期，第17位代表性别，奇数为男，偶数为女）。

3、编写程序，计算如下数列的值：1-2+3-4+…-966。其中，所有数字为整数，从1开始递增，奇数为正，偶数为负。

4、在我国古代《算经》里有一个著名的不定方程问题：鸡翁一值钱五，鸡母一值钱三，鸡雏一值钱一。百钱买百鸡，问鸡翁、鸡母、鸡雏各几何？编写程序，求解此问题并输出解。

【实验内容】

1. 程序清单

**（1）判断回文数**

**思路：**python没有直接反转int型整数的方法，而字符串有现成的方法，即reversed()。注意：函数返回反向的迭代器对象；此方法并不会修改原来序列的元素顺序。

**代码和注释：**

num = input("请输入一个五位数字")

newstr = ''.join(reversed(num)) # 将迭代器转为字符串

# 两个字符串进行比较

if newstr == num:

    print("此数是回文数")

else:

print("此数不是回文数")

**（2）输出出生年月日**

**思路：**程序须根据身份证号计算年龄，为使程序更具一般性，考虑使用time类来获取当前的时间。记得将所有数字都转为整型，原因见注释。

**代码和注释：**

import time # 导入时间类

t = time.localtime() # 获取当前时间

id = input("请输入18位合法身份证号：")

# 此处转为整数的原因是和现在的年份相减

year = int(id[6] + id[7] + id[8] + id[9])

age = t.tm\_year - year # 当前年份 - 出生年份

# 此处转为整数的原因是需要忽略开头的0

m = int(id[10] + id[11]) # 出生月

# 此处转为整数的原因是需要忽略开头的0

d = int(id[12] + id[13]) # 出生日

# 此处转为整数的原因是判断男女

sex = int(id[16])

if sex % 2:

    sex = "男"

else:

    sex = "女"

print(f"出生年月：{year}年{m}月{d}日")

print(f"年龄：{age}岁")

print(f"性别：{sex}")

**（3）计算数列**

**思路：**方法一：使用if分支结构；方法二：使用 (-1)^n 计算正负号。此处采用方法二。

**代码和注释：**

sum = 0

for i in range(1, 967):

    sum += (-1) \*\* (i - 1) \* i # 采用 (-1)^n 计算正负号

print(sum)

**（4）不定方程**

**思路：**列方程，利用二重循环求所有解：

设鸡翁x只、鸡母y只、鸡雏z只，

根据题意，5x + 3y + z / 3 = 100，x + y + z = 100，两者联立，得7x + 4y = 100

**代码和注释：**

for x in range(0, 15):

    for y in range(0, 26):

        if (7 \* x + 4 \* y == 100):

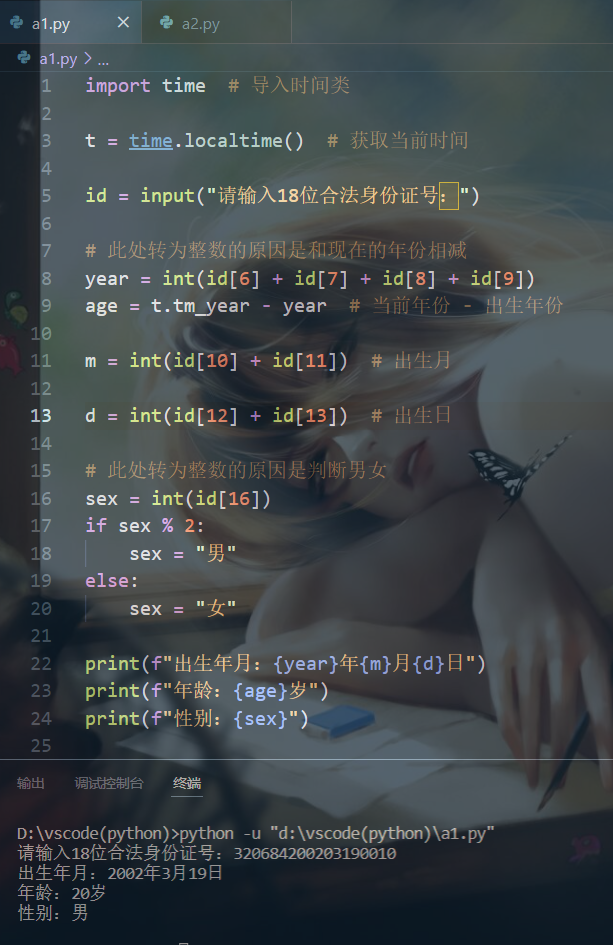
            print(f"鸡翁{x}只，鸡母{y}只，鸡雏{100 - x - y}只")

1. 结果截图

**（1）判断回文数**



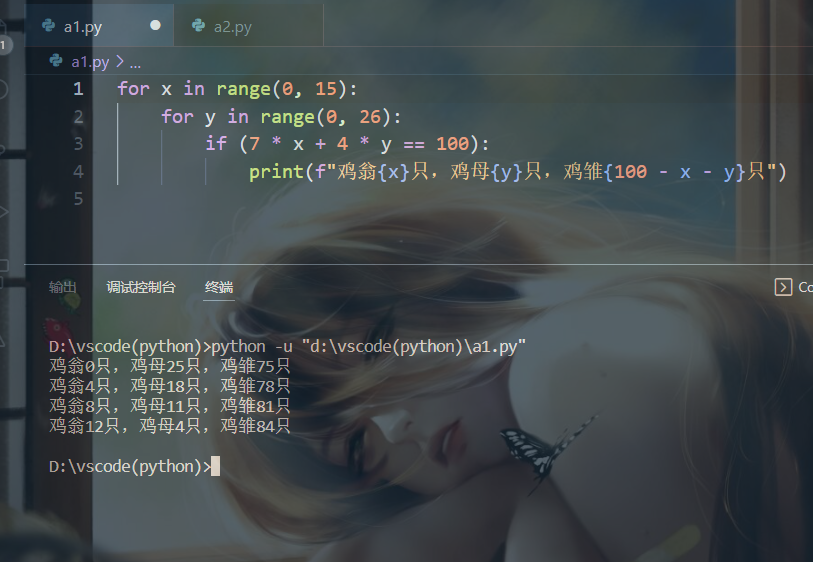
**（2）输出出生年月日**



**（3）计算数列**



**（4）不定方程**



【实验体会】

第一题使我熟悉了对字符串的反转操作；第二题了解了python中time库的使用方法，并熟悉了对于字符串中单个字符的访问方式；第三题熟悉了循环和乘方的使用；第四题熟悉了二重循环的使用。