山东大学网络空间安全学院

Python高级程序设计 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：202100460065 | 姓名：李昕 | | 班级：密码 21 二班 |
| 实验题目：Python选择与循环 | | | |
| 实验学时：2 | | 实验日期：2022.10.4 | |
| 实验目的：熟悉条件与循环控制语句 | | | |
| 硬件环境：  AMD Ryzen 7 5800H with Radeon Graphics     3.20 GHz  机带 RAM 16.0 GB (13.9 GB 可用) | | | |
| 软件环境：  操作系统：windows 11  编译器：IDLE | | | |
| 实验步骤与内容：  本次实验共有5个问题：  1.编写程序，判断今天是今年的第几天；  2.编写程序实现如下功能：循环遍历i从0到4，j从0到4，k从0到4，如果i、j、k都等于3时，输出“it’s over”，结束整个循环，否则打印i、j、k的值；  3.编写代码，实现如下功能：循环遍历i从0到4，j从0到4，k从0到4，打印除i==j==k==3之外的i、j、k的值；  4.编写程序，使用蒙特·卡罗方法计算圆周率近似值（提示，import random,使用随机数模拟坐标）；  5.暴力破解MD5 值：给定一个MD5值的十六进制串  '23eeeb4347bdd26bfc6b7ee9a3b755dd'，该MD5值对应的明文为一个5-10个英文字符/数字的字符串，返回对应的明文（提示: from hashlib import md5. Python中的hashlib模块提供了很多常见hash算法的实现。）。  【判断日期】  该问题可以分为两个子问题：  1.可以通过系统内置模块或者手动输入两种方式得到今天的日期  2. 通过内置模块得到日期时，导入time模块，使用strftime函数，并将获得的字符串赋值给变量a，使用split函数将年、月、日分开，通过判断是否是闰年，来进行天数的相加，如果是闰年则2月为29天。  **代码实现：**  通过系统内置模块、函数得到今天的日期：   1. def date1(): 2. import time 3. a=time.strftime("%Y/%m/%d") 4. shuju=a.split('/') 5. m=int(shuju[0]) 6. n=0 7. if (m%100==0)&(m%400==0): *#判断是否是闰年* 8. n=1 9. elif(m%100!=0)&(m%4==0): 10. n=1 11. aDict1={1:31,2:28,3:31,4:30,5:31,6:30,7:31,8:31,9:30,10:31,11:30,12:31} 12. aDict2={1:31,2:29,3:31,4:30,5:31,6:30,7:31,8:31,9:30,10:31,11:30,12:31} 13. i=1 14. sum=0 15. if n==1: 16. while i<int(shuju[1]): 17. sum=sum+aDict2[i] 18. i=i+1 19. else: 20. while i<int(shuju[1]): 21. sum=sum+aDict1[i] 22. i=i+1 23. sum=sum+int(shuju[2]) 24. print('今天是第',sum,'天') 25. date1()   或者通过手动输入日期实现：   1. def date(a): *#计算第几天* 2. shuju=a.split('/') 3. m=int(shuju[0]) 4. n=0 5. if (m%100==0)&(m%400==0): *#判断是否是闰年* 6. n=1 7. elif(m%100!=0)&(m%4==0): 8. n=1 9. aDict1={1:31,2:28,3:31,4:30,5:31,6:30,7:31,8:31,9:30,10:31,11:30,12:31} 10. aDict2={1:31,2:29,3:31,4:30,5:31,6:30,7:31,8:31,9:30,10:31,11:30,12:31} 11. i=1 12. sum=0 13. if n==1: 14. while i<int(shuju[1]): 15. sum=sum+aDict2[i] 16. i=i+1 17. else: 18. while i<int(shuju[1]): 19. sum=sum+aDict1[i] 20. i=i+1 21. sum=sum+int(shuju[2]) 22. print(a,'是第',sum,'天') 23. a=input('请以/为间隔符输入日期，省略无效的0：') 24. date(a)   运行结果：    【循环遍历实验2，3】  1.由分析可知，实验2，3两个嵌套循环，在结构上类似，但在终止循环条件判断等方面有微小差异；  2.第一个嵌套循环要求在i=j=k=3的时候，终止循环，打印“it’s over”并将除i=j=k=3之外的所有情况输出。实现该功能的循环时，当循环遍历到i=j=k=3时使用**break**将三层循环退出；  3.第二个嵌套循环要求循环遍历i从0到4，j从0到4，k从0到4，打印除i==j==k==3之外的i、j、k的值。实现该功能的循环时，当循环遍历到i=j=k=3时，使用continue退出单层循环，不打印此时的结果。  **代码实现1：**   1. def sduxh(): *#循环遍历* 2. tag=True 3. for i in range(5): 4. if(tag==False): 5. break 6. for j in range(5): 7. if(tag==False): 8. break 9. for k in range(5): 10. if(i==3 & j==3 & k==3): 11. print("it's over") 12. tag=False 13. break 14. else: 15. print(i,j,k) 16. sduxh()   **代码实现2：**   1. def xsduxhplus(): *#循环遍历不输出（3，3，3）* 2. for i in range(5): 3. for j in range(5): 4. for k in range(5): 5. if(i==3 & j==3 & k==3): 6. continue 7. else: 8. print(i,j,k) 9. sduxhplus()   两个运行结果：      【计算圆周率近似值】  该问题可以分解为三个子问题：  1.通过随机数生成横纵坐标  2. 通过设置判断条件判断生成随机点是否在圆的区域内  3.设置投掷飞镖次数（由于是概率问题，次数要尽可能大来减小误差）  代码实现：   1. def pi\_(a): *#蒙特卡罗方法计算圆周率近似值* 2. from random import random 3. n=0 4. for i in range(a): 5. x=random() 6. y=random() 7. if(x\*x+y\*y<1): 8. n=n+1 9. print("{:.10f}".format(4\*n/a)) 10. b=input('请输入扔飞镖的次数：') 11. pi\_(int(b))   运行结果：    【暴力破解MD5】  将该问题划分为两步：  1.首先通过迭代的方法，在permutations中导入itertools，用x in permutations（’x’,6）生成迭代对象。  2.然后利用字符串的方法: ‘’.join()来转换为字符串类型，然后生成该字符串的md5编码，最后与已知的md5码进行字符串比较，得到结果。  代码实现：   1. from hashlib import md5 2. from itertools import permutations 3. m1 = md5()    *# 构造hash对象* 4. for item in permutations('qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM1234567890',6): 5. s1=''.join(item) 6. m1 = md5(s1.encode()) 7. if m1.hexdigest()=='23eeeb4347bdd26bfc6b7ee9a3b755dd': 8. print(s1)   执行结果： | | | |
| 结论分析与体会：  通过本次实验，我熟悉了Python中循环与条件控制语句的编程方法，掌握了循环中break和continue的用法，并对密码算法md5的加密以及解密方式进行了学习，并尝试暴力破解md5，验证了md5具有较好的安全性。 | | | |