

# 苏州大学实验报告

院、系	计算机学院	年级专业	计算机科学	姓名	柯骅	学号	2027405033
课程名称	Python 程序设计					成绩	
指导教师	李正华	同组实验者		实验日期			

## 实验名称 实验七 元组、字典与集合

### 一. 实验目的

通过本次实验要达到如下目的：

1. 掌握元组、字典和集合两种数据结构的基本概念
2. 掌握元组的常用操作和常用方法，并理解相应的编程应用过程
3. 掌握字典的常用操作和常用方法，并理解相应的编程应用过程
4. 掌握集合的常用操作和常用方法，并理解相应的编程应用过程

### 二. 实验内容

1. 【元组】现在 8 名体检人员的体重信息如下：

(65.5, 70.2, 100.5, 45.5, 88.8, 55.5, 73.5, 67.8)，请编写程序计算出方差。

2. 【字典】使用元组编写程序，模拟评委打分的过程。评委打分标准：去掉 1 个最高分和 1 个最低分，计算平均分。
3. 【字典】从键盘上随机输入若干个大写英文字母，编写程序使用字典统计所输入的每个字母出现的次数。
4. 【字典】在程序中创建两个字典，找出并显示两个字典中相同的键。
5. 【字典】在程序中创建两个字典，找出并显示两个字典中具有相同值（要求数据类型也相同）的键。
6. 【字典】创建一个有关雇员姓名和编号处理的程序。从键盘输入一组雇员姓名和编号。在此基础上实现：
  - a) 按照雇员姓名的顺序输出数据，雇员姓名显示在前面，后面是对应的雇员编号。
  - b) 按照雇员编号的顺序输出数据，雇员编号显示在前面，后面是对应的雇员姓名。
7. 【集合】小明想在学校中请一些同学一起做一项问卷调查，请编程帮助小明解决如下问题：
  - a) 用户输入 N；
  - b) 为了实验的客观性先用计算机生成 N 个 1~1000 之间的随机整数( $N \leq 1000$ )；
  - c) 对于其中重复的数字，只保留一个，将其余相同的数字去掉，不同的数对应着不同学生的学号；
  - d) 然后再将这些数从小到大排序，按照排好的顺序去找同学做调查。输出排序的结果。

8. 【集合】通过[0,500]范围内随机数发生的方法分别创建两个整数数据的集合，要求每个集合中数据的个数分别要超过 200 个。在此基础上实现：
- a) 求出两个集合中不相同的数据，并进行显示。要求每行显示 10 条，每个数占 5 列，右对齐；
  - b) 求出两个集合中相同的数据，并进行显示。要求每行显示 10 条，每个数占 5 列，右对齐；
9. 【集合】使用 random 模块生成一个整数类型的随机数集合：从 0 到 9(包括 9)中随机选择，生成 1 到 10 个[0,1000]范围内的随机数。这些数字组成集合 A。同理，按此方法生成集合 B。在此基础上实现以下功能：
- a) 显示 A 和 B 的结果。要求每行最多显示 10 个数，每个数占 5 列，右对齐；
  - b) 要求用户输入 A|B 和 A & B 的结果，并告诉用户他(或她)的答案是否正确。如果用户回答错误，允许他(或她)修改解决方案，然后重新验证用户输入的答案。如果用户三次提交的答案均不正确，程序将显示正确结果。

### 三. 实验步骤和结果（每一道题必须画流程图）

1. 第一题程序如下：

```
a=(65.5, 70.2, 100.5, 45.5, 88.8, 55.5, 73.5, 67.8)
p=sum(a)/8#平均值
for i in a:
    tot=abs(p-i)**2#计算
print('%.5f'%(tot/8))
```

#思路：由于是八个固定的数，所以用元组  
# 用方差公式和循环计算即可

运行结果如下：

1.21096

2. 第二题程序如下：

```
l=list(map(int,input('please input the socres:').split()))
l.sort() #在列表时就对它进行排序较为便捷
t=tuple(l) #转换成元组才可以进行剥离元素
a,*b,c=t #剥离最大最小值
print(sum(b)/len(b))
```

#思路：采用元组可以剥离的特点可以快速去掉最大最小值

运行结果如下：

```
please input the socres:5 4 2 3 6 1 9 8 7
5.0
```

3. 第三题程序如下:

```
d={}
s=input('please input a string:')
for i in s:#枚举字符串中的字母
    d[i]=d.get(i,0)#用 get 防止报错
    d[i]+=1#统计
for i in d.items():
    print('%s:%d'%(i[0],i[1]))
#思路: 用字典来实现对同一种字母的统计
#      将字母作为 key, 次数作为 value
```

运行结果如下: please input a string:ASDFSDFGDFGH  
A:1  
S:2  
D:3  
F:3  
G:2  
H:1

4. 第四题程序如下:

```
a={'A':3,'#':10,3:100,(1,2):1}
b={'B':4,'#':50,1:999,(1,2):1}#在程序中创建两个字典
c=[]#储存最终结果
for i in a.keys():
    if i in b.keys():
        c.append(i)
print('the same keys are:',c)

#思路: 只要枚举其中一个字典的 keys 的值
#      再判断在另一个之中有没有即可
```

运行结果如下:

the same keys are: ['#', (1, 2)]

5. 第五题程序如下:

```
def f(s):#在 a 和 b 的键值中寻找值为 s 的键
    x,y=[],[]
    for i in a.keys():
        if a[i]==s:
            x.append(i)
    for i in b.keys():
        if b[i]==s:
            y.append(i)
    return x,y

a={'A':3,'#':10,3:100,(1,2):1}
```

```
b={'B':100,'#':10,1:999,(1,2):10}
c=set(a.values())#a, b 中都要有一样的值，所以只需要转化并枚举其中一个
```

```
for i in c:
    x,y=f(i)
    if x != [] and y != []:#输出
        print('have the same value:',i,'in a:',x)
        print('have the same value:',i,'in b:',y)
```

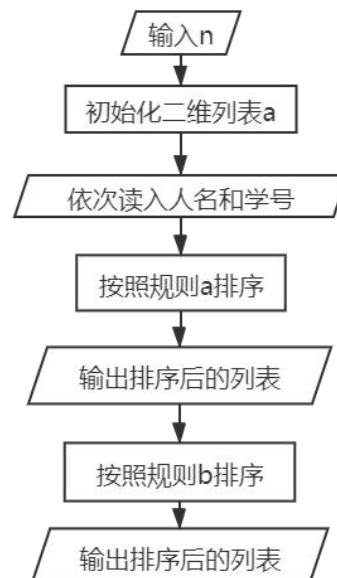
#思路：用 f 函数返回在 a 和 b 的键值中为 s 的键  
# 组成两个数组  
# 判断他们是否都不为空，若是，则输出

运行结果如下：

```
have the same value: 10 in a: ['#']
have the same value: 10 in b: ['#', (1, 2)]
have the same value: 100 in a: [3]
have the same value: 100 in b: ['B']
```

6. 第六题程序如下：

```
n=int(input('please input the number of staffs:'))
a=[[None for _ in range(2)] for _ in range(n)]
for i in range(n):
    a[i][0],a[i][1]=list(input('please input name and number').split())
#根据 a 规则进行排序
a.sort(key=lambda x:x[0])
#输出 a 规则结果
print('a:')
for i in range(n):
    print(a[i][0],a[i][1])
#根据 b 规则进行排序
a.sort(key=lambda x:x[1])
#输出 b 规则结果
print('b:')
for i in range(n):
    print(a[i][1],a[i][0])
```



#思路：采用 sort 方法中的关键字调用（key=lambda）分别排序输出即可

运行结果如下：

```
please input the number of staffs:3
please input name and numbera 3
please input name and numberb 1
please input name and numberc 2
a):
a 3
b 1
c 2
b):
1 b
2 c
3 a
```

7. 第七题程序如下：

```
import random
n=int(input('please input the number of numbers:'))
s=set()
for i in range(n):
    s.add(random.randint(1,1000))#在集合中增加一个 1 到 1000 的随机数
l=list(s)#转化成列表方便排序
l.sort()
print(l)#输出
```

#思路：（1）按题目要求创建出集合  
# （2）转化成列表排序

运行结果如下：

```
please input the number of numbers:10
[4, 67, 100, 115, 235, 320, 385, 597, 709, 952]
```

8. 第八题程序如下：

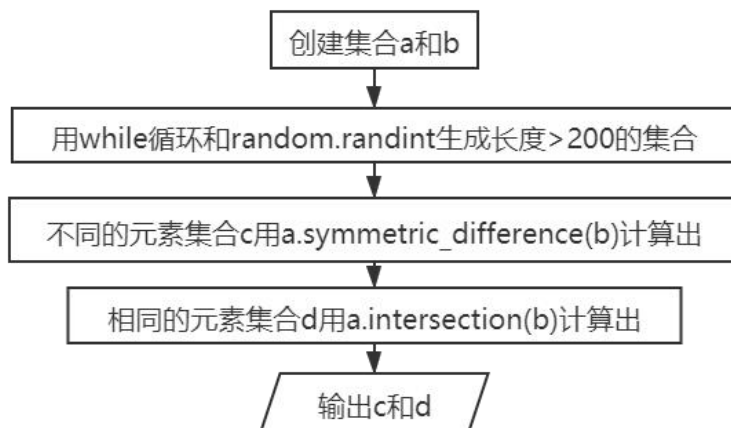
```
def p(s):#用函数输出，避免写两遍
    for i in range(len(s)):
        if (i + 1) % 5 == 0:
            print('%5d' % s[i])
        else:
            print('%5d' % s[i], end='')
import random
#生成 a 和 b 两个集合
a,b=set(),set()
while len(a)<=200:
    a.add(random.randint(0,500))
while len(b)<=200:
```

```

        b.add(random.randint(0,500))
#不同的
c=list(a.symmetric_difference(b))
#相同的
d=list(a.intersection(b))
print('the different nums are:')
p(c)
print('the same nums are:')
p(d)

```

#思路: (1) 用 random 生出随机数  
# (2) 用 symmetric\_difference() 和 intersection() 来找同和不同



运行结果如下:

the different nums are:	221	222	224	225	226	445	447	450	451	454
1 2 6 7 10	228	230	232	233	235	455	458	467	471	472
12 15 17 18 19	237	241	243	244	245	473	477	478	483	485
20 23 24 28 29	246	247	248	249	250	488	489	493	494	500
30 33 34 39 42	253	258	260	261	262					
43 44 45 46 47	264	267	268	270	271	the same nums are:				
48 54 55 56 59	273	274	277	279	281	256	4	388	263	137
62 63 64 66 69	282	283	284	286	287	11	139	141	269	397
76 78 81 82 83	288	289	291	295	300	398	146	276	21	150
85 86 87 89 90	301	304	306	307	308	27	285	413	414	160
97 100 102 104 106	310	311	313	316	318	416	417	165	167	423
109 114 117 118 120	320	321	322	328	329	429	303	431	49	50
124 125 128 129 130	331	332	333	334	335	305	180	309	437	57
131 134 135 136 143	336	338	339	342	343	186	314	315	317	190
144 145 149 152 153	347	348	349	350	355	191	446	65	448	196
154 155 157 158 159	356	360	361	362	365	325	453	71	200	456
162 163 164 166 168	366	371	372	373	374	77	461	79	337	466
170 172 173 176 177	377	381	382	383	386	341	469	345	475	223
178 179 181 182 184	387	389	393	394	395	479	98	354	482	486
188 189 193 194 195	396	400	402	403	404	105	234	490	108	492
198 201 202 204 205	405	406	408	409	410	111	239	119	376	379
206 207 208 210 213	411	412	418	420	422	127				
215 217 218 219 220	427	428	430	432	435					

9. 第九题程序如下:

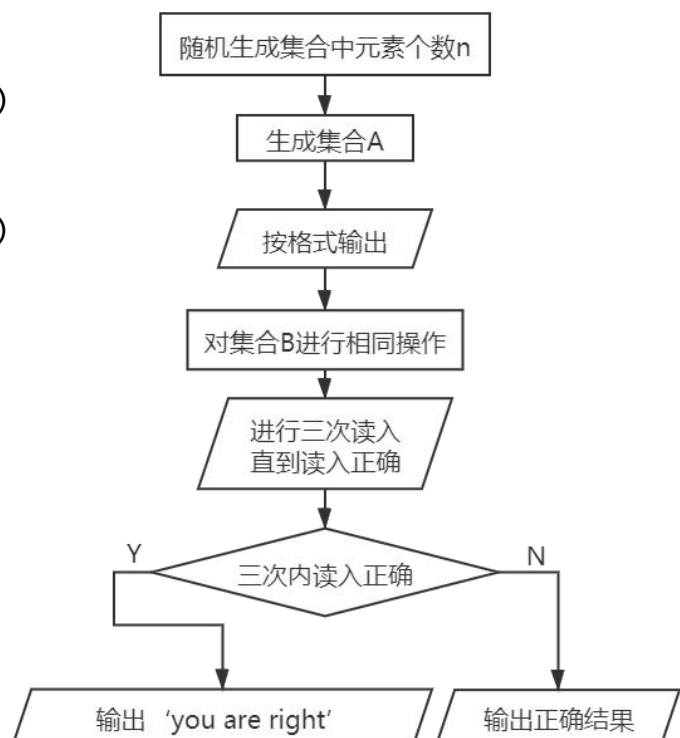
```
import random
def g():
    n=random.randint(1,10)
    x=set()
    for i in range(n):
        x.add(random.randint(0,1000))
    return x

#读入 A|B 和 A&B 并返回结果
def ff():
    global bing
    global jiao
    for i in range(3):
        x = list(map(int, input('please input A|B:').split()))
        y = list(map(int, input('please input A&B:').split()))
        if x==bing and jiao==y:
            return 'Binggo, you are right'
        else:
            print('Sorry, you are wrong,please try again')
    return('the right answer : \nA|B:%s\nA&B:%s'%(bing,jiao))

a,b=g(),g()#用函数生成 ab 集合
bing,jiao=a|b,a&b#生成交集和并集
#转化成列表方便下面的处理
a,b,bing,jiao=list(a),list(b),list(bing),list(jiao)
print('A is :')
for i in range(len(a)):
    print('%5d'%a[i],end='')
print('\nB is :')
for i in range(len(b)):
    print('%5d'%b[i],end='')
print()

print(ff())
```

#思路: (1) 随机生成 AB  
# (2) 计算 A|B 和 A&B  
# (3) 读入并判断





运行结果如下：

```
A is :  
    824  113  979  
B is :  
    364  
please input A|B:0  
please input A&B:0  
Sorry, you are wrong,please try again  
please input A|B:0  
please input A&B:0  
Sorry, you are wrong,please try again  
please input A|B:0  
please input A&B:0  
Sorry, you are wrong,please try again  
the right answer :  
A|B:{824, 113, 979, 364}  
A&B:set()
```

```
A is :  
    395  933  
B is :  
    88  642  
please input A|B:0  
please input A&B:0  
Sorry, you are wrong,please try again  
please input A|B:395 933 642 88  
please input A&B:  
Binggo, you are right
```



拓展题：

(1) 角谷猜想

程序如下：

```
n=int(input())
#while 循环判断是否结束
while n!=1:
    if n%2==1:
        print(n, '*3+1=', n*3+1, sep=' ')
        n=n*3+1
    else:
        print(n, '/2=', n//2, sep=' ')
        n=n//2
```

#思路：

#如果是奇数，则乘 3 加 1

#如果是偶数，则除以 2

#得到的结果再按照上述规则重复处理

#最终总能够得到 1

运行结果如下：

40

40/2=20

20/2=10

10/2=5

5\*3+1=16

16/2=8

8/2=4

4/2=2

2/2=1

21

21\*3+1=64

64/2=32

32/2=16

16/2=8

8/2=4

4/2=2

2/2=1

## （2）辗转相除法求最大公约数

程序如下：

```
x,y=list(map(int,input("请输入两个正整数:").split()))
a,b=max(x,y),min(x,y)#a 是较大值, b 是较小值
while a%b!=0:
    temp=a%b
    a=b
    b=temp#交替过程
print("最大公约数是:",b)
```

#思路：辗转相除法最后得到 b 是最大公约数

运行结果如下：

```
请输入两个正整数:12 8
最大公约数是: 4
```

```
请输入两个正整数:99 2
最大公约数是: 1
```

## 四. 实验总结（包括对老师的建议）

掌握了元组、字典和集合两种数据结构的基本概念

掌握了元组的常用操作和常用方法，并理解了相应的编程应用过程

掌握了字典的常用操作和常用方法，并理解了相应的编程应用过程

掌握了集合的常用操作和常用方法，并理解了相应的编程应用过程

理解了 python 中集合的求并集与交集等等方法，能在具体问题中熟练应用

对函数有了进一步的了解，函数在 python 中有非常灵活的用法

了解了角谷猜想和辗转相除法

其中，辗转相除法为求最大公约数提供了一种时间复杂度很低的方法，避免了枚举所带来的耗时