

数组复习

本文中提到的实现方法基本都是考试常考或者最古老（常规）的方法，仅供参考。

数组的创建和基本操作

1. 随机产生10位同学的身高数据（[150,180]）存储在数组a中，计算10位同学的平均身高。

```
1 import random
2 a=[random.randint(150,180) for i in range(10)]
3 Sum=0
4 for i in range(len(a)):
5     Sum+=a[i]
6 print(Sum/len(a))
```

2. 查找最大/最小值

在数组a=[9,15,8,23,3]中输出最大值，以及最大值的下标位置。

```
1 a=[9,15,8,23,3]
2 MaxIndex=0
3 for i in range(len(a)):
4     if a[i]>a[MaxIndex]:
5         MaxIndex=i
6 print(a[MaxIndex],MaxIndex)
```

进阶：在数组a=[[6,4,7],[3,5,8]]中，输出最小值以及最小值的下标位置？

```
1 a=[[6,4,7],[3,5,8]]
2 MaxIndex=[0,0]
3 for i in range(len(a)):
4     for j in range(len(a[i])):
5         if a[i][j]>a[MaxIndex[0]][MaxIndex[1]]:
6             MaxIndex[0]=i
7             MaxIndex[1]=j
8 print(a[MaxIndex[0]][MaxIndex[1]])
9 print(MaxIndex[0],MaxIndex[1])
```

3. 在一系列整数中，不在第一个也不在最后一个，但比左、右数都大的数就是相

邻最大数。请输出所有相邻最大数。例如：数组[1, 3, 2, 4, 1, 5, 3, 9, 7, 10, 8, 23, 85, 43]，则输出：3 4 5 9 10 85

```
1 a=[1, 3, 2, 4, 1, 5, 3, 9, 7, 10, 8, 23, 85, 43]
2 for i in range(1,len(a)-1):
3     if a[i]>a[i-1] and a[i]>a[i+1]:
4         print(a[i],end=' ')
```

数组的访问

4. 已知数组a=[1,2,3,4,5,6,7,8]，将数据第1个元素与最后1个元素交换，第2个与倒数第2个交换，以此类推。

```
1 a=[1,2,3,4,5,6,7,8]
2 b=[]
3 n=len(a)
4 for i in range(1,n-1):
5     if a[i]>a[i-1] and a[i]>a[i+1]:
6         b.append(a[i])
7         print(a[i],end=' ')
8 print(b)
```

5. 【输出数组中所有大于key的元素中的最小值】。例如：a=[12,7,9,13,18,2],所有大于10的元素中最小值为12。

```
1 a=[12,7,9,13,18,2]
2 Min=0
3 flag=True
4 n=len(a)
5 for i in range(n):
6     if a[i]>10:
7         if flag or a[i]<Min:
8             Min=a[i]
9             flag=False
10 print(Min)
```

6. 【数组最大连续子序列和】。例如：a=[-2,11,-4,13,-5,-2],最大连续子序列和为20，即为a[1]+a[2]+a[3]。

```
1 a=[-2,11,-4,13,-5,-2]
2 MaxSum=0
3 n=len(a)
4 for i in range(n-1):
5     Sum=0
6     for j in range(i,n):
7         Sum+=a[j]
8         if Sum>MaxSum:
9             MaxSum=Sum
10 print(MaxSum)
```

7. 【数组最长连续上升元素个数】。例如a=[2,1,3,7,5,8]，最长连续上升元素个数为3，即a[1]、a[2]、a[3]。

方案一

```
1 a=[2, 1, 3, 7, 5, 8]
2 b=[]
3 n=len(a)
4 StartP=0
5 EndP=1
6 while EndP<n:
7     if a[EndP]>a[EndP-1]:
8         EndP+=1
9     else:
10        b.append(a[StartP:EndP])
11        StartP=EndP
12        EndP+=1
13 b.append(a[StartP:EndP])
14 MaxLen=0
15 for i in range(len(b)):
16     if len(b[i])>MaxLen:
17         MaxLen=len(b[i])
18 print(MaxLen)
19 print(b)
```

方案二

```
1 a=[2, 1, 3, 7, 5, 8]
2 n=len(a)
3 StartP=0
4 EndP=1
5 MaxLen=0
6 while EndP<n:
7     if a[EndP]>a[EndP-1]:
8         EndP+=1
9     else:
10        if EndP-StartP>MaxLen:
11            MaxLen=EndP-StartP
12        print(a[StartP:EndP])
```

```

13     StartP=EndP
14     EndP+=1
15 if EndP-StartP>MaxLen:
16     MaxLen=EndP-StartP
17 print(MaxLen)

```

8. 求一批整数中出现最多的数字，例如输入3个整数1234,2345,34456，其中出现次数最多的数字是4，出现了4次。如果有多个数字的次数相同，如1234，2345，3456，3和4均出现了3次，此时输出数字较大的那个数，即4。

```

1 a=[1234,2345,34456]
2 b=[0]*10
3 for i in range(len(a)):
4     k=a[i]
5     while k>0:
6         b[k%10]+=1
7         k=k//10
8 print(b)
9 Max=0
10 for i in range(len(b)):
11     if b[i]>b[Max]:
12         Max=i
13 print(Max)

```

数组元素的插入

9. 【有序数组中插入新数据】。例如：a=[2,5,6,9,11,20,21],在该数组中插入新数据x（比如x=10）后，数组依然有序。

```

1 a=[2,5,6,9,11,20,21]
2 x=10
3 for i in range(len(a)):
4     if a[i]>x:
5         a.append(a[-1])
6         for j in range(len(a)-1,i,-1):
7             a[j]=a[j-1]
8         a[i]=x
9         break
10 print(a)

```

数组元素的删除

10. 【数组中删除指定位置的元素】。例如数组a=[2,5,6,9,11]，删除元素a[k]。如果k=2，则删除后数组元素为a=[2,5,9,11]。

```

1 a=[2,5,6,9,11]
2 k=2
3 for i in range(k,len(a)-1):
4     a[i]=a[i+1]
5 a.pop()
6 print(a)

```

11. 删除非降序数组a=[1,2,2,3,3]中重复的元素，只要确保删除后数组前3个元素为：[1,2,3，。。。]即可。

```

1 a=[1,1,1,2,2,3,3,4,4,4,5]
2 i=0
3 while i<len(a):
4     if a[i] not in a[:i]:
5         i+=1
6     else:
7         for j in range(i,len(a)-1):

```

```

8 |         a[j]=a[j+1]
9 |     a.pop()
10 | print(a)

```

数组的嵌套

12. 模拟骰子随机投掷程序。用数组a存储骰子每次出现的数字（1—6），假设随机投掷了20次，编写程序实现每个数字（1-6）出现的次数。提示：ct[1]的值表示数字1出现的次数。

```

1 | import random
2 | ct=[0]*7
3 | n=20
4 | for i in range(n):
5 |     r=random.randint(1,6)
6 |     ct[r]+=1
7 | for i in range(1,7):
8 |     print(i,":",ct[i])

```

13. 下列程序的功能是产生10个[0,30]范围内的随机整数，并统计这10个随机整数中不同整数的个数（重复出现算1个）。例如数组[9,1,7,9,1,2,9,2,1,2]中不同整数的个数为4。

```

1 | import random
2 | a=[]
3 | n=10
4 | Num=0
5 | for i in range(n):
6 |     r=random.randint(0,30)
7 |     print(r,end=' ')
8 |     if r not in a:
9 |         a.append(r)
10 |        Num+=1
11 | print()
12 | print(Num)

```

数组的合并

14. 两个有序数组合并成新的有序数组。合并两个升序的数组，将数组b的数据插入到数组a中，要求最后得到的数组a中的数据仍是升序排列。a=[1,3,6,18],b=[5,9,12],合并后a=[1,3,5,6,9,12,18]

```

1 | a=[1,3,6,18]
2 | b=[5,9,12,20,22,24]
3 | aP=0
4 | bP=0
5 | while aP<len(a) and bP<len(b):
6 |     if a[aP]<b[bP]:
7 |         aP+=1
8 |     else:
9 |         a.append(a[-1]) #括号里写啥都行
10 |        for i in range(len(a)-1,aP,-1):
11 |            a[i]=a[i-1]
12 |        a[aP]=b[bP]
13 |        bP+=1
14 | if bP!=len(b):
15 |     for i in range(bP,len(b)):
16 |         a.append(b[i])
17 | print(a)

```

15. 两个有序数组合并成新的有的数组。比如：a=[2,5,6,9,11],b=[1,10,12,13,15,18],合并后c=[1,2,5,6,9,10,11,12,13,15,18]

```
1 a=[2,5,6,9,11]
2 b=[1,10,12,13,15,18]
3 c=[]
4 aP=0
5 bP=0
6 while aP<len(a) and bP<len(b):
7     if a[aP]>b[bP]:
8         c.append(b[bP])
9         bP+=1
10    else:
11        c.append(a[aP])
12        aP+=1
13 print(c)
```

by Delt-HT