

2022 年春季沙河联盟《大学计算机基础》实验指导书

实验 2 程序控制结构与列表

编程之前请认真阅读实验指导，了解题目要求，明确编程思路。

当在本地运行程序成功、并使用实验指导书给出的**所有**输入样例和输出样例检查无误后，再将源代码提交到在线实验平台（OJ 系统，在线评测系统）进行自动测试，以便平台自动记录你的成绩。

请在规定时间内完成所有题目并确保在 OJ 系统上通过测试，测试结果为“AC”。

在线实验平台（OJ 系统）链接：<https://www.comthinking.cn/login>

代码提交截止时间：本周四（2022 年 4 月 14 日）23: 55。

本次实验要求按照“实验报告模板”撰写实验报告。请将实验报告和本次实验所有源代码放在以“**学校缩写+学号+姓名**”命名的文件夹中，压缩后在 OJ 的“实验报告”栏目提交。截止时间为**下周四（2022 年 4 月 21 日）23: 55**。

1. 实验目的

- (1) 加深对顺序结构、选择结构、循环结构的理解。学会分析问题选择合适的程序控制结构。
- (2) 学会用 Python 编写 for 循环，while 循环。
- (3) 学会用 Python 编写 if 多重条件判断。
- (4) 学会用 Python 编写循环嵌套结构。
- (5) 进一步掌握输出格式控制方法。
- (6) 掌握 Python 基本的数据结构：列表的使用。能够熟练应用列表的常用方法去处理这种数据结构，并能利用它们解决相应的问题。

实验任务 2-1 破译密电

题目描述：

这天，小 Z 给你发来了一串密电，并希望你能破译出它。

一串密电由 n 个整数组成，其中每对相邻的数 a, b （位置分别为 $2*i-1$ 和 $2*i$ ）

表示一个含 a 个整数 b 的列表。破译后的密电由所有这样的列表按顺序拼接而成。

小 Z 不仅希望你能破译出密电，还希望你能告诉她破译后的密电中最大的数是多少，以及它出现了多少次，并把这两个数附在破译出的密电序列后。你能告诉她正确的结果吗？

输入：

输入一行 n 个整数 ($2 \leq n \leq 1000$ 且 n 为偶数)，各数由单个空格分隔，表示原始密电。

输出：

输出一行若干整数，各数由单个空格分隔，表示破译后的密电序列，且在最后附加密电序列的最大值和其出现次数。

输入样例 1：

1 2 3 4 5 6

输出样例 1：

2 4 4 4 6 6 6 6 6 5

输入样例 2：

4 7 8 5 2 3 6 9

输出样例 2：

7 7 7 7 5 5 5 5 5 5 5 3 3 9 9 9 9 9 9 6

实验指导：

1. 样例 1 解释

输入“1 2 3 4 5 6”中数对 1 2 表示列表中包含 1 个 2；数对 3 4 表示列表中包含 3 个 4；数对 5 6 表示列表中包含 5 个 6。三个列表拼接起来，为[2, 4, 4, 4, 6, 6, 6, 6, 6]，表示破译后的密电。其中，最大的数为 6，它出现了 5 次。将这两个数附在破译出的密电序列后，故得到输出为 2 4 4 4 6 6 6 6 6 5。

2. 使用 `input().split()` 将一行整数作为列表读入。

- 1) 列表是 Python 最基本的数据结构之一，列表是一个有序的可修改的元素序列，其中的元素可以是任何变量，元素的顺序由下标(0, 1, 2, 3, 4……)表示。
- 2) 类比于一列火车（列表），有着很多车厢，每一节车厢里可以装任意的货物（变量），车厢相互之间有顺序，每一节车厢都可拆卸，可添加，可更换，车厢的编号为 0, 1, 2, 3……。即第一节车厢编号 0，

第二节编号 1，以此类推。

3. 列表操作参考

1) `lis.append(x)` 在列表 `lis` 末尾添加元素 `x`。

比如：`list1=[1,2]`，`list1.append(1)`，此时 `list1=[1,2,1]`。

2) `lis.count(x)` 返回列表 `lis` 中 `x` 出现的个数。

比如：`list1=[1,2,2,3]`，则 `list1.count(1)=1`，`list1.count(2)=2`，`list1.count(3)=1`。

3) `print(*lis)` 将列表 `lis` 中的元素输出，并使不同元素之间以单个空格分隔。

3. `max(lis)`可以返回 `lis` 中的最大值。

实验任务 2-2 $3n+1$ 问题

题目描述：

卡拉兹在 1950 年的世界数学家大会上公布了一个猜想，对任何一个大于 1 的自然数 n ，如果它是偶数，那么把它除以 2；如果它是奇数，那么把它变成 $(3n+1)$ 。这样一直变换下去，最后一定在某一步得到 $n=1$ 。例如：

$3 \rightarrow 3 \times 3 + 1 = 10 \rightarrow 10 / 2 = 5 \rightarrow 5 \times 3 + 1 = 16 \rightarrow 16 / 2 = 8 \rightarrow 8 / 2 = 4 \rightarrow 4 / 2 = 2 \rightarrow 2 / 2 = 1$

现在输入一个大于 1 的整数，请你用计算机模拟出它变换到 1 的过程。

输入：

一行，一个大于 1 的正整数 n 。

输出：

若干行，每行一个整数，模拟 n 经过多次变换后变为 1 的过程。

输入样例 1：

3

输出样例 1：

3

10

5

16

8

4
2
1

输入样例 2:

16

输出样例 2:

16
8
4
2
1

样例 1 解释:

3 -> $3*3+1=10$ -> $10/2=5$ -> $5*3+1=16$ -> $16/2=8$ -> $8/2=4$ -> $4/2=2$ -> $2/2=1$ 。

实验目的:

本题主要考查对 while 语句的应用。

实验指导:

1. Python 编程中 while 语句用于循环执行程序，即在某条件下，循环执行某段程序，以处理需要重复处理的相同任务。其基本形式为：

```
while 循环条件(condition):  
    执行语句(statements)  
.....
```

本题在 $n>1$ 时会不断变换 n 的值，直至 $n=1$ ，因此 while 中的循环条件为 $n>1$ 。

2. while 中的执行语句为对 n 进行的转换，若 n 为奇数则 n 变为 $3*n+1$ ，否则 n 变为 $n/2$ ，这里考查对 if - else 语句的应用。
3. Python 中的 求余 (%) 运算 $a\%b$ 表示 a 除以 b 的余数，可在本题中用来判断 n 的奇偶性。
4. 注意 Python 中除法 $/$ 运算与整除 $//$ 运算的区别。除法运算会将 n 的类型由整型变为浮点型。

思考: “如果它是偶数，那么把它除以 2”——这里应该采用 $n/2$ 还是 $n//2$?

5. 该问题又被称为冰雹猜想，在完成程序后，你可以试试 27 这个数字。虽然它是一个貌不惊人的自然数，但是如果按照上述方法进行运算，则它的上浮下沉异常剧烈：首先，27 要经过 77 个步骤的变换到达峰值 9232，然后又经过 34 个步骤到达谷底值 1。全部的变换过程需要 111 步之多。

实验任务 2-3 GPA 计算

题目描述：

GPA，或称平均学分绩点，是用来衡量学生学习成果的重要指标。这里的 GPA 采用 4 分制（即满分为 4 分），算法如下：

设某门课程的百分制成绩为 x ，则相应的 $GPA = 4 - \frac{3 \times (100 - x)^2}{1600} (60 \leq x \leq 100)$,

当分数为 60 分时 GPA 为 1，60 分以下 GPA 为 0。

现输入 $N (1 \leq N \leq 100)$ 门课的百分制成绩 x_1, x_2, \dots, x_N 和每门课对应的学分 h_1, h_2, \dots, h_N 。

通过各门课 GPA 计算总 GPA 的公式为：

$$\text{总 GPA} = \frac{GPA_1 h_1 + GPA_2 h_2 + \dots + GPA_N h_N}{h_1 + h_2 + \dots + h_N}。$$

请编写程序计算总 GPA，将结果保留到小数点后 2 位。

输入：

输入数据为 $N + 1$ 行。

第 1 行为一个正整数 N ，表示获得百分制成绩的课程门数；

第 2 行至 $N + 1$ 行为 N 门课程的百分制成绩和其对应的学分，第 i 行为第 i 门科目对应的百分制成绩和学分 x_i, h_i ，成绩与学分之间用一个空格隔开。

输出：

输出数据包含一行，为计算出的总 GPA(保留到小数点后 2 位)。

输入样例 1：

```
2
85 3
60 2
```

输出样例 1：

```
2.55
```

数据范围

$$0 \leq x_i \leq 100, x_i \in Z; 0.5 \leq h_i \leq 6, h_i = 0.5k (k \in Z)$$

实验指导：

1. 注意计算单科 GPA 时，如果成绩低于 60 分 则 GPA 为 0；
2. 使用 `a, b = map(float, input().split())` 将输入转换为浮点数；
3. 对于多组数据，可以使用for循环来控制读取数据的次数以及对数据处理的次数。

实验任务 2-4 小 Y 同学坐地铁

题目描述：

在清明假期时，小 Y 同学所在的学校终于解封了，小 Y 同学安排了三天的出游计划，交通方式都选择乘坐地铁。同时小 Y 同学每次乘坐都购买了单程票。

但是假期结束后，有同学提醒小 Y 同学，地铁可以购买单日票和三日票。地铁的单日票价格为 15 元，购买后单日不限次数免费乘坐；三日票价格为 40 元，购买后三日内不限次数免费乘坐。

小 Y 意识到，如果自己合理选择购买单日票和三日票，可以让自己达到最少的花费，小 Y 同学十分悔恨没有提前知道这一点，现在小 Y 同学想知道，如果自己用最合适的方式购买地铁票，能比自己每次都购买单程票少花多少钱，你能帮他算算吗？

输入：

输入共三行。第 i 行代表第 i 天小 Y 同学购买地铁票的情况。

每行有若干个以空格分隔的数 代表小 Y 同学在这一天购买的单程票的价格。

输出：

输出一行。代表小 Y 同学最多能少花多少钱。

输入样例 1：

5 10 7

3 3 3

5 5

输出样例 1：

7

输入样例 2：

20

10 10

7 8 9

输出样例 2:

24

样例 1 解释:

小 Y 每次都购买单程票, 总共花费了 $5+10+7+3+3+3+5+5=41$ 元。最优的方案应该是第一天购买单程票, 之后两天购买单程票, 共花费 $15+(3+3+3)+(5+5)=34$ 元, 故采用最优方案比每次都购买单程票能少花费 $41-34=7$ 元。

实验指导:

1. Python 提供了一个内建函数 `sum()`, 可以计算一个列表中所有数的和, 当前前提是列表中的元素是可求和的。如 `lis=[2, 3, 5]`, 则 `sum(lis)` 会返回 10。
2. 设计思路
 - (1) 采用 for 循环: 计算每一天所有单程票的价格之和, 并与单程票价格相比较, 取较小值; 累加计算三天的最少花费之和。
 - (2) 如果三天最少花费之和大于三日票的价格, 则选择购买三日票。

实验任务 2-5 拍照 (选做)

题目描述:

喜多川同学今天正在参加 cosplay 展, 现在准备拍合照!

对于一张合照, cosplayer 们会排成一行, 如果对于任意至少有两个 cosplayer 的连续片段, 男性的数量都不超过女性的数量, 我们则称这张合影是美丽的。

现在已有 n 位 cosplayer 排成了一行, 我们用一个二进制的字符串表示, 其中的 0 代表男性, 1 代表女性。为了保证合影是美丽的, 我们可以在任意位置邀请一些额外的 cosplayer (可能为 0)。请注意您不能从队伍中移走任何 cosplayer。

现在喜多川同学想知道最少需要邀请几位 cosplayer 才能让合影是美丽的, 你能帮帮她吗?

输入:

输入两行。

第一行为一个正整数 n ，表示初始状态下队伍中 cosplayer 的数量。

第二行为一个长度为 n 的二进制字符串，表示初始状态下队伍中 cosplayer 的性别，其中 0 代表男性，1 代表女性。

输出：

输出一行。

表示最少需要邀请的 cosplayer 的数量。

输入样例 1：

3

000

输出样例 1：

4

输入样例 2：

4

0110

输出样例 2：

0

输入样例 3：

19

1010110000100000101

输出样例 3：

17

数据范围：

$1 \leq n \leq 100000$ 。

样例 1 解释：

可以向初始队列 000 中插入 4 位 cosplayer 变为 0110110，此时即可保证任意一段人数不少于 2 的连续片段中，男性的数量不多于女性。

实验指导：

1. Python 字符串

Python 字符串可以通过索引访问每个字符串中的元素，如 $s = \text{'abccxy'}$ ，则有 $s[0] = \text{'a'}$, $s[1] = \text{'b'}$, $s[2] = \text{'c'}$, $s[3] = \text{'c'}$ 以此类推。

2. 主要设计思路

- (1) 考虑初始队伍中如果只有两个男性，你需要如何解决这个问题，至少需要邀请几位 cosplayer?
- (2) 考虑整个队伍中两两相邻的男性之间的距离。