



RC.CODE.PYTHON 赛项说明

1 赛项信息

1.1 赛项内容

- A. 赛项将全面检验选手基于 Python 软件编程语言的技术实现能力，鼓励创新、培养实践能力和解决问题的能力。
- B. 赛项均为个人参与。
- C. 赛项题目设置均为客观题+编程题。
- D. 赛项各组别均为使用电脑在规定平台参与答题。
- E. 赛项时长均为 90 分钟。

1.2 组别设置

编程项目	编程语言	组别划分	年龄要求
Python 编程	Python 编程语言	Python - A 组	≤12 岁
	Python 编程语言	Python - B 组	≤18 岁

1.3 赛项题目

项目组别	单选题	编程题	项目时间
Python - A 组	5 道题，共 100 分	5 道题，共 300 分	90 分钟
Python - B 组	5 道题，共 100 分	5 道题，共 300 分	90 分钟

2 规则和得分

2.1 比赛规则

- A. 比赛要求选手在规定平台使用编码的方式，完成指定题目作答，答题过程中禁止切出编译器，禁止打开其他软件和网页，否则一律视为作弊，取消成绩。
- B. 每位选手只有一次挑战机会，规定时间未登录规定平台的选手视为放弃挑战。
- C. 准备阶段，选手须按照指定平台规定步骤完成作答准备，并自主操作进入正式答题。
- D. 挑战期间，选手不得离开电脑答题区域。
- E. 挑战期间，选手不得抄袭他人、不得作弊、不得直接与其他选手的电脑直接接触，如有违反，该选手记 0 分。
- F. 挑战过程中，不得求助他人，不得干扰其他选手备赛和答题，不得损坏公用设备，一经发现，勒令退出。
- G. 挑战期间，电脑上不得开启任何通讯软件，如有违反，该选手记 0 分。
- H. 挑战期间，选手不得使用手机、电话手表等通信、摄影电子设备及外接存储设备，如



有违反，取消挑战资格。

- I. 本规则的解释权归 RC 组委会所有。

2.2 赛项得分

根据题目完成情况以及完成度和时间综合评定，完成题目数量越多、完成度越高且用时较短的选手成绩越高。

3 设备要求

- A. 自备电脑，电脑操作系统：Mac OS、Windows 10 或以上操作系统；浏览器采用谷歌浏览器（69.0 版本以上）、Firefox、Internet Explorer 11 以上，推荐 Chrome。
- B. 自备智能手机一部。

4 大纲要求

4.1 Python - A 组大纲

- A. 输入与输出：
 - a. 概念理解。
 - b. 使用基本输入输出和简单运算为主的标准函数。
- B. 掌握注释的方法
- C. 基础数据类型
 - a. 字符串、数值、布尔值的概念。
 - b. 数据类型的表示方法。
 - c. 数值类型与字符串类型之间的转换方法
- D. 变量
 - a. 概念理解。
 - b. 变量赋值。
 - c. 变量的使用方法。
- E. 条件语句
 - a. if-else 的使用方法。
 - b. if-elif-else 的使用方法
- F. 运算符
 - a. 比较运算符的表示和使用方法。
 - b. 逻辑运算符的表示和使用方法。
- G. 循环结构
 - a. 无限循环 (while True) 的使用。
 - b. 使用 break 跳出循环结构。
 - c. for 循环的使用方法。
- H. 掌握随机数的概念和使用方法。
- I. 能用编程实现四则运算和四则混合运算。



- J. 掌握列表的创建、索引、增加、删除、查找、修改、遍历等常用操作方法。
- K. 掌握字典的创建、增加、删除、查找、修改、遍历等常用操作方法。
- L. 函数
 - a. 掌握函数的定义和调用方法。
 - b. 掌握参数定义和返回值的定义与使用。
- M. 使用不同算法解决各类常见问题。

4.2 Python - B组大纲

- A. 输入与输出：
 - a. 概念理解。
 - b. 使用使用基本输入输出和简单运算为主的标准函数。
- B. 掌握注释的方法
- C. 基础数据类型
 - a. 字符串、数值、布尔值的概念。
 - b. 数据类型的表示方法。
 - c. 数值类型与字符串类型之间的转换方法
- D. 变量
 - a. 概念理解。
 - b. 变量赋值。
 - c. 变量的使用方法。
- E. 条件语句
 - a. if-else 的使用方法。
 - b. if-elif-else 的使用方法
- F. 运算符
 - a. 比较运算符的表示和使用方法。
 - b. 逻辑运算符的表示和使用方法。
- G. 循环结构
 - a. 无限循环 (while True) 的使用。
 - b. 使用 break 跳出循环结构。
 - c. for 循环的使用方法。
- H. 掌握程序的缩进规则。
- I. 掌握随机数的概念和使用方法。
- J. 能用编程实现四则运算和四则混合运算。
- K. 掌握列表的创建、索引、增加、删除、查找、修改、遍历等常用操作方法。
- L. 掌握字典的创建、增加、删除、查找、修改、遍历等常用操作方法。
- M. 函数
 - a. 掌握函数的定义和调用方法。
 - b. 掌握参数定义和返回值的定义与使用。
- N. 使用不同算法解决各类常见问题。



- O. 掌握 range()方法的使用。
- P. 掌握二维列表的索引查找元素的方法。
- Q. 掌握字典、列表的嵌套运用。
- R. 掌握基本事件处理模块的使用。

5 RC.CODE.Python 样题示例

【Python - A 组】样题示例

一、选择题(每题 20 分)

1、在计算机中，二进制数系统使用哪两个数字？ ()

- A、0 和 1
- B、0 和 2
- C、1 和 2
- D、1 和 10

答案 A

2、若元组 $y = ('A', 'B', 'C', 'D')$ ，则能够返回元素 'A' 的语句是 ()

- A、 $y[-3]$
- B、 $y[1]$
- C、 $y[0]$
- D、 $y[4]$

答案 C

3、初始时 $x = 5$ ， $x // 2 + x ** 2$ 运算后的值为 ()

- A、26.5
- B、26
- C、27.5
- D、27

答案 D

4、以下程序的最终打印结果 ()

```
l = [1, 2, 3, 4, 5, 6, 7, 8, 9]
print ( l [2 : 4] )
```



- A、 [2 , 3]
- B、 [1 , 2 , 3 , 4]
- C、 [3 , 4]
- D、 [2 , 3 , 4]

答案 C

5、执行以下程序，打印结果为 ()

```
L = []  
for letter in 'Python':  
    if letter == 'n':  
        continue  
    L.append(letter)  
print(L)
```

- A、 []
- B、 ['P', 'y', 't', 'h']
- C、 ['P', 'y', 't', 'h', 'o']
- D、 ['o', 'n']

答案 C

二、编程题 (总分 300 分)

1、编程题一 (40 分)

编程实现:

给定一个小写字母，输出其对应的大写字母。

输入描述:

输入一个小写字母。

输出描述:

输出小写字母对应的大写字母。

样例输入:

a

样例输出:

A



评分标准:

10分: 能正确输出一组数据;

10分: 能正确输出两组数据;

10分: 能正确输出三组数据;

10分: 能正确输出四组数据。

2、编程题二 (40分)

编程实现:

输入一个正整数 $N(4 < N < 40)$, 然后打印出 N 排 N 列星号(*)。

输入描述:

输入一个正整数 N 。

输出描述:

打印出 N 排 N 列 "*"。

样例输入:

5

样例输出:

```
*****  
*****  
*****  
*****  
*****
```

评分标准:

10分: 能正确输出一组数据;

10分: 能正确输出两组数据;

10分: 能正确输出三组数据;

10分: 能正确输出四组数据。

3、编程题三 (60分)

编程实现:

给定一个正整数 N , 和 M 个不同的正整数。然后将 M 个正整数中任意两个数进行组合, 且求出每组组合的和, 问 M 个正整数中有多少组组合的和恰好等于 N 。



如：正整数 N 为 6, M 为 5, 5 个不同的正整数分别为 1, 2, 3, 4, 5。

任意两数组合有 10 组: 1+2, 1+3, 1+4, 1+5, 2+3, 2+4, 2+5, 3+4, 3+5, 4+5。

其中和正好等于 6 的组合有 2 组: 1+5, 2+4。

输入描述:

第一行输入一个正整数 N 。

第二行输入 M 个不同的正整数, 且正整数之间以一个英文逗号隔开。

输出描述:

输出 M 个不同正整数中有多少组组合的和恰好等于 N 。

样例输入:

5

1,2,3,4,5

样例输出:

2

评分标准:

15 分: 能正确输出一组数据;

15 分: 能正确输出两组数据;

15 分: 能正确输出三组数据;

15 分: 能正确输出四组数据。

4、编程题四 (60 分)

编程实现:

N 个小朋友围成一圈 ($N \leq 60$), 并按照顺序进行编号 (编号 1 到 N), 然后从编号为 1 的小朋友开始持续报数, 当报到的数字中含 3 或者数字是 3 的倍数的小朋友, 不能报数字而是报“过” (例如 3, 6, 13, 31 这些数字都报“过”)。

一次游戏中, 小朋友们玩的不亦乐乎, 突然有小朋友发现前边报过的数字中已经有报错的, 而后边小朋友还在继续报数。请你帮助小朋友找出第一次报错的小朋友是编号几。

例如: 有 3 个小朋友, 编号为 1, 2, 3。编号 1 的小朋友报 1、编号 2 的小朋友报 2、编号 3 的小朋友报“过”、编号 1 的小朋友报 4、编号 2 的小朋友报 5、编号 3 的小朋友报 6、编号 1 的小朋友报 7; 此时第一次报错的小朋友为编号 3。(因为编号 3 的小朋友应该报“过”时, 报成了“6”, 而“6”是 3 的倍数)



输入描述:

第一行输入一个正整数 N, 表示小朋友的人数。

第二行输入一组有错的报数序列 (“0” 表示含 3 或者是 3 的倍数的数字), 序列元素之间以一个英文逗号隔开。

输出描述:

输出第一次报错的小朋友是编号几。

样例输入:

3
1,2,0,4,5,6,7

样例输出:

3

评分标准:

15 分: 能正确输出一组数据;

15 分: 能正确输出两组数据;

15 分: 能正确输出三组数据;

15 分: 能正确输出四组数据。

5、编程题五 (100 分)

编程实现:

小马需要将 N 件物品从河的一岸搬运到河的另一岸, 每次搬运的物品为 1 到 3 件。请问小马将 N 件物品全部搬运过去有多少种方案。

例如: N=3, 将 3 件物品全部搬运过去有 4 种方案:

方案一: 第一次搬运 1 件, 第二次搬运 1 件, 第三次搬运 1 件;

方案二: 第一次搬运 1 件, 第二次搬运 2 件;

方案三: 第一次搬运 2 件, 第二次搬运 1 件;

方案四: 一次搬运 3 件。

输入描述:

输入一个正整数 N, 表示需要搬运的物品数。

输出描述:

输出将 N 件物品全部搬运过去有多少种方案。



样例输入：

3

样例输出：

4

评分标准：

25分：能正确输出一组数据；

25分：能正确输出两组数据；

25分：能正确输出三组数据；

25分：能正确输出四组数据。

【Python - B 组】样题示例

一、选择题 (每题 20 分, 总分 100 分)

1. 下列表达式中, 返回 True 的是 ()

- A、 $3 != 3$
- B、 $5 > 4 > 3$
- C、 $8 \% 2 == 1$
- D、True and False

答案 B

2. 以下关于函数描述正确的是 ()

- A、函数的定义必须在程序的开头
- B、函数定义后需要调用才能执行
- C、函数体与关键字 def 必须左对齐
- D、函数定义后, 其程序可以自动执行

答案 B

3. 下面的函数用于判断什么? ()

```
def is_prime(num):  
    for i in range(2, num):  
        if num % i == 0:  
            return False
```



return True

- A、加法计算
- B、判断质数
- C、循环索引
- D、排序列表

答案 B

4、下列表达式在遍历字典 d 时，变量 x 代表字典值的是 ()

- A、for x in d:
- B、for x in d.keys():
- C、for x in d.items():
- D、for x in d.values():

答案 D

5、已知 t = ("bian" , "cheng" , "sai")，执行 t[::-1]语句输出的结果是 ()

- A、("sai" , "cheng" , "bian")
- B、["sai" , "cheng" , "bian"]
- C、{ "sai" , "cheng" , "bian" }
- D、运行报错

答案 A

二、编程题 (总分 300 分)

1、编程题一 (40 分)

编程实现:

输入长方形的长和宽，输出长方形面积。

输入描述:

输入长方形的长和宽 (中间用逗号隔开)。

输出描述:

输出长方形的面积。

样例输入:

3 4

样例输出:

12



评分标准:

10分: 能正确输出一组数据;

10分: 能正确输出两组数据;

10分: 能正确输出三组数据;

10分: 能正确输出四组数据。

2、编程题二 (40分)

编程实现:

给定一个四位正整数, 然后将其倒序输出。

输入描述:

输入一个个位数不为“0”四位正整数。

输出描述:

将这个正整数倒序输出。

样例输入:

1234

样例输出:

4321

评分标准:

10分: 能正确输出一组数据;

10分: 能正确输出两组数据;

10分: 能正确输出三组数据;

10分: 能正确输出四组数据。

3、编程题三 (60分)

编程实现:

有 N 个正整数, 现对 N 个正整数进行不同方式的排列, 每次排列后都会按照以下规则进行一次计算, 例如: 排列方式不同, 最后计算出的结果也不相同。

计算规则:

第一次: 第一个数乘以第二个数乘以第三个数, 结果记录为 M(1);

第二次: 第二个数乘以第三个数乘以第四个数, 结果记录为 M(2);



RobotChallenge 世界机器人盛会

第三次：第三个数乘以第四个数乘以第五个数，结果记录为 $M(3)$ ；

第 $N-2$ 次：第 $N-2$ 个数乘以第 $N-1$ 个数乘以第 N 个数，结果记录为 $M(N-2)$ 。

最后计算 $M(1)+M(2)+M(3)+\dots+M(N-2)$ 的数值。

找出一种排列方式使这个数值最大。

例如： $N=4$ ，4 个正整数分别为 1, 2, 3, 4，那么排列方式就会有 24 种：

其中排列方式为 1, 3, 4, 2 时，按照规则计算 2 次： $1*3*4=12$ ， $3*4*2=24$ ；乘积相加： $12+24=36$ 。

这种排序方式是所有乘积相加的数值最大为 36。

输入描述：

输入 N 个正整数 ($3 \leq N$)，正整数之间一个英文逗号隔开

输出描述：

找出所有乘积相加的数值最大的排列方式，并输出数值

样例输入：

1,2,3,4

样例输出：

36

评分标准：

15 分：能正确输出一组数据；

15 分：能正确输出两组数据；

15 分：能正确输出三组数据；

15 分：能正确输出四组数据。

4、编程题四 (60 分)

编程实现：

给出一排黑色带数字的小球（数字为 0 到 9），和一排白色带数字的小球（数字为 0 到 9），现从两排小球中一共选取 K 个小球排成一排，使得 K 个小球排成新的一排组成的数字最大。

例如：

黑色小球的原顺序为：

②⑤③

白色小球的原顺序为：

⑥②④①

K 为 3；



RobotChallenge 世界机器人盛会

从两排小球中共选取 3 个小球，排成⑥⑤④。可以组成的最大数字为 654。

输入描述：

第一行输入一组正整数，代表黑色小球，每个正整数范围为 0 到 9，正整数之间以一个英文逗号隔开

第二行输入一组正整数，代表白色小球，每个正整数范围为 0 到 9，正整数之间以一个英文逗号隔开

第三行输入一个正整数 K (K 小于等于所有小球的总数)，表示从所有小球中共选取 K 个小球

输出描述：

输出一个整数，表示按照要求选取 K 个小球后，组成的最大数字

样例输入：

2,5,3

6,2,4,1

3

样例输出：

654

评分标准：

15 分：能正确输出一组数据；

15 分：能正确输出两组数据；

15 分：能正确输出三组数据；

15 分：能正确输出四组数据。

5、编程题五 (100 分)

编程实现：

目前有一个包含 n 个正整数的序列 a 。将计划对序列进行多次升序排序，每次升序排序都要选择一个区间 $[l, r]$ (l, r) 并对区间内所有数字，即 a_l, a_{l+1}, \dots, a_r 进行升序排序。每次升序排序会在上一次升序排序的结果上进行。现在请你计算出多次升序排序后的序列。

输入描述：

第一行包含一个正整数 n ，含义如题面所示

第二行包含 n 个正整数 a_1, a_2, \dots, a_n ，代表序列。第三行包含一个正整数 q ，代表排序次数之后 q 行，每行包含两个正整数 l_i, r_i ，代表将区间 $[l_i, r_i]$ 内所有数字进行



升序排序

输出描述:

输出一行包含 n 个正整数, 代表多次升序排序后的序列

样例输入:

5

3 4 5 2 1

3

4 5

3 4

1 3

样例输出:

1 3 4 5 2

评分标准:

25分: 能正确输出一组数据;

25分: 能正确输出两组数据;

25分: 能正确输出三组数据;

25分: 能正确输出四组数据。