

蓝桥杯省赛 (Python)

十二届蓝桥杯Python组省赛试题

一、选择题

第一题

设 `s = 'Hello Lan Qiao '`, 执行 `print(s[4:11])` 输出的结果为 ()。

*选择题严禁使用程序验证

- A、lo Lan Qi B、lo Lan Q C、o Lan Qi D、o Lan Q

第二题

循环语句 `for i in range(8,-4,-2):` 执行了几次循环 ()。

*选择题严禁使用程序验证

- A、4 B、5 C、6 D、7

第三题

导入 `random` 模块后, 执行 `print(random.randrange(10, 100, 2))` 语句, 输出的结果可能是 ()。

*选择题严禁使用程序验证

- A、1 B、24 C、63 D、100

第四题

执行以下程序, 输出的结果是 ()。

*选择题严禁使用程序验证

```
height = {'小蓝':80,'小青':100,'小圆':70}
```

```
height['小青'] = 70
```

```
print(sum(height.values()))
```

- A、220 B、210 C、250 D、70

第五题

导入 `turtle` 模块后, 执行 `turtle.circle(80,steps=6)` 语句, 所绘制的图形以下说法正确的是 ()。D

*选择题严禁使用程序验证

- A、绘制出一个半径为 80 像素的圆 B、绘制出六个半径为 80 像素的圆
C、绘制出一个半径为 80 像素的圆和一个六边形 D、绘制出一个半径为 80 像素的圆内接正六边形

选择题答案在文档最后一页

二、编程题

第一题

(注: input()输入函数的括号中不允许添加任何信息)

编程实现:

给定一个正整数N, 计算N除以7的商。

输入描述: 输入一个正整数N

输出描述: 输出N除以7的商 (商为整数)

样例输入: 8

样例输出: 1

第二题

(注: input()输入函数的括号中不允许添加任何信息)

编程实现:

给定一个正整数N, 计算出1到N之间所有奇数的和。

输入描述: 输入一个正整数N

输出描述: 输出1到N之间 (包含1和N) 所有奇数的和

样例输入: 5

样例输出: 9

第三题

(注: input()输入函数的括号中不允许添加任何信息)

提示信息:

“水仙花数”是指一个三位正整数, 其各位数字立方的和等于该数本身。

例如: 153是一个“水仙花数”, 因为 $1^3 + 5^3 + 3^3$ 等于153。

编程实现:

给定一个正整数N, 判断100到N之间有多少个“水仙花数”。

输入描述: 输入一个正整数N($100 < N < 1000$)

输出描述: 输出100到N之间 (包含100和N) 有多少个“水仙花数”

样例输入: 160

样例输出: 1

第四题

(注: input()输入函数的括号中不允许添加任何信息)

编程实现:

有一组连续正整数, 随机乱序后生成一组数据后, 小蓝不小心删掉了其中一个数, 已知所删掉的这个数不是这组数据中最小的也不是最大的, 现在请你编写程序帮助小蓝找到删除的那个数。

输入描述:

按照“编程实现”中的描述模仿输入一组这样的正整数数(正整数之间以英文逗号隔开), 在输入的时候少一个数(这个数不是这组数据中最小的也不是最大的), 这个数作为小蓝删除掉的那个数, 且加上小蓝删除的那个数组数据是连续的

输出描述: 输出删除掉的是哪个数

样例输入: 3,2,4,6,7

样例输出: 5

第五题

(注: input()输入函数的括号中不允许添加任何信息)

提示信息:

有一个密室逃脱游戏, 有100间密室连在一排。密室编号是从1开始连续排列一直排到第100间密室, 如下图:

1	2	3	4	5	6	99	100
---	---	---	---	---	---	-----	-----	----	-----

游戏规则:

1. 玩家初始位置在1号密室;
2. 每次玩家可以进入右边的一个密室, 也可以跳过一个密室进入下个密室(如: 当玩家当前在3号密室, 他可以进入4号密室也可以进入5号密室);
3. 有毒气的密室不能进入需要避开。

编程实现:

给定三个正整数X, Y, M ($X < Y < M \leq 100$), 表示三个密室编号。X号密室和Y号密室有毒气泄漏, 不能进入, 玩家需要进入到M号密室。按照游戏规则进入M号密室有多少种路线方案。

例如: X=2, Y=4, M=7, 进入M号密室有2种路线方案, 分别是1->3->5->6->7路线和1->3->5->7路线。

输入描述: 输入三个正整数X, Y, M ($X < Y < M$), X和Y表示有毒气密室编号, M表示需要进入的密室编号, 且三个正整数之间以英文逗号隔开

输出描述: 输出进入M号密室有多少种路线方案

样例输入: 2,4,7

样例输出: 2

第六题

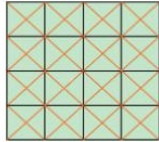
(注: input()输入函数的括号中不允许添加任何信息)

编程实现:

有一个 $N \times N$ 的矩阵方格和 N 个棋子, 现在需要将 N 个棋子按要求放置到矩阵方格中。

要求如下:

- 1.任意两个棋子不能在同一行
- 2.任意两个棋子不能在同一列
- 3.任意两个棋子不能在同一对角线上(下图红色线段都为对角线)



根据以上要求, 问 N 个棋子放置到 $N \times N$ 矩阵方格中有多少种放置方案

例如: 4×4 的矩阵方格, 4个棋子, 有2种放置方案

	*		
			*
*			
		*	

输入描述: 输入一个正整数 $N(1 < N < 11)$, 表示一个 $N \times N$ 的矩阵方格和 N 个棋子数量

输出描述: 输出 N 个棋子按要求放置到 $N \times N$ 的矩阵方格中有多少种放置方案

样例输入: 4

样例输出: 2

选择题答案: DCBAD