

# 2022 年第 20 届 NOC Python 赛项决赛

## 小高组

### 选择题

第 1 题

以下选项的关键字中，不属于分支结构的是？

- A. elif
- B. if
- C. else
- D. in

答案: D

第 2 题

关于 Python 的数据类型，下列选项描述错误的是？

- A. 10.0 是一个浮点数
- B. 字符串可以保存在变量中，也可以单独存在
- C. int('1.0')返回的结果是整数
- D. 布尔类型的值只有 True 和 False

答案: C

第 3 题

已知字符串 s = 'hello python' (注意中间有一个空格)，想要截取其中的 'python'，选项中不能实现的是？

- A. s[6:]
- B. s[6: 11]
- C. s[-6: 12]
- D. s[-6:]

答案: B

第 4 题

下列选项的输出结果为 True 的是？

- A. 2>0 and 3<0 or 3<0
- B. 2>0 or 3<0 and 3<0
- C. not ('1' == 1) and 1 != 1.0 or '2' == '2.0'
- D. ('test' == 'Test') or 1 != 1.0 and 2 == 2.0

答案: B

第 5 题

下列代码执行完毕后，输出的结果是？

```
nums = [3, 1, 2, 4, 0]
nums.insert(1, 0)
```

```
nums.pop()
nums.append(1)
nums.remove(2)
print(nums)
A. [0, 3, 1, 4, 1]
B. [0, 1, 2, 4, 1]
C. [3, 0, 1, 4, 1]
D. [3, 0, 2, 4, 1]
```

答案: C

#### 第 6 题

以下选项能正确随机一个 6~8 范围内整数的是?

```
A. random.randint(6, 9)
B. randint.random(6, 9)
C. random.randint(6, 8)
D. random.choice(6, 8)
```

答案: C

#### 第 7 题

请问执行完以下代码后, 海龟画布上画出的线段长度为多少?

```
import turtle as t
t.speed(0)
t.color("red")
t.pensize(3)
t.shape("circle")
t.forward(100)
t.forward(-10)
t.forward(20)
t.backward(20)
t.backward(-10)
t.left(90)
t.home()
t.right(90)
```

```
A. 120
B. 110
C. 100
D. 90
```

答案: B

#### 第 8 题

请问执行完以下代码后, 输出结果为?

```
tot = 0
for n in range(3, 66 + 1):
    if n % 2 == 0:
```

```
        continue
    if n % 10 == 9:
        break
    tot += n
print(tot)
```

- A. 18
- B. 33
- C. 2208
- D. 15

答案: D

### 第 9 题

请问执行完以下代码后, 输出结果为?

```
def is_lucky(num):
    if num % 6 == 0:
        return True
    while num > 0:
        if num % 10 == 6 or num % 10 == 8:
            return True
        num /= 10
    return False
```

```
ls = [13, 666, 888, 32, 332, 987, 555, 345, 876]
```

```
cnt = 0
for n in ls:
    if is_lucky(n):
        cnt += 1
print(cnt)
```

- A. 3
- B. 4
- C. 5
- D. 6

答案: A

### 第 10 题

请问执行完以下代码后, 输出结果为?

```
ls = [1, 2, 3, 4, 5, 1]
n = len(ls)
for i in range(n):
    for j in range(i + 1, n):
        if ls[j] > ls[i]:
            tmp = ls[i]
            ls[i] = ls[j]
```

```
        ls[j] = tmp
print(ls)
A. [1, 2, 3, 4, 5, 1]
B. [1, 1, 2, 3, 4, 5]
C. [5, 4, 3, 2, 1, 1]
D. 报错, 无输出
答案: C
```

## 程序题

1、给定列表求最值

```
ls = [-1, -3, -345, -22, -2, -999]
# 设置最大值计数器 maxn 初值
maxn = _____
for n in _____:
# 满足什么条件就更新 maxn 呢?
    if _____:
        maxn = n
# 输出最大值
print(maxn)
```

参考代码:

```
ls = [-1, -3, -345, -22, -2, -999]
maxn = ls[0]
for n in ls:
    if n > maxn:
        maxn = n
print(maxn)
```

2、列表函数

有两个列表 A 和 B, 使用列表 C 来获取两个列表中共有的元素。

例如: A = [1, 'a', 4, 90] B = ['a', 9, 'j', 1], 则 C =

```
[1, 'a']
A = [1, 'a', 4, 90]
B = ['a', 9, 'j', 1]
C = []
for i in range(0, len(A)):
    for j in range(____):
        if A[i] == ____:
            _____
print(C)
```

参考代码:

```
A = [1, 'a', 4, 90]
B = ['a', 9, 'j', 1]
C = []
for i in range(len(A)):
    for j in range(len(B)):
        if A[i] == B[j]:
            C.append(B[j])
print(C)
```

### 3、吉利又霸气的英文名

每个字母都有自己的积分：它在字母表是第几个它的积分就是几，比如：**a**的积分是**1**，**z**的积分是**26**。一个字符串的积分为所有字符的积分之和。我们知道中国人特别喜欢**6**和**9**，觉得**6**是一个幸运数字，而**9**则是一个霸气的数字，毕竟九五至尊嘛~若一个字符串的积分既是**6**的倍数，又是**9**的倍数，则此字符串为吉利又霸气的字符串。现在给你一个英文名，请你来判断这个英文名是不是吉利又霸气，如果是的，请输出'**Yes**'，否则，请输出'**No**'。

【输入格式】

一行，一个字符串，表示要判断的英文名，均为小写字母。

【输出格式】

一行，表示判断结果，如果输入的英文名是吉利且霸气的，则输出'**Yes**'，否则，输出'**No**'。

【输入样例 1】

mick

【输出样例 1】

Yes

【输入样例 2】

allen

【输出样例 2】

No

参考代码:

```
name = input()
s = '0abcdefghijklmnopqrstuvwxyz'
tot = 0
for c in name:
    tot += s.find(c)

if tot % 6 == 0 and tot % 9 == 0:
    print('Yes')
else:
```

```
print('No')
```

#### 4、掷骰子

班级每个周五下课都会进行一次大扫除，阿明和阿强是一组，这次他们的任务是擦玻璃和打扫厕所，他们都不愿意去打扫厕所，于是聪明的阿明决定用一个掷骰子的游戏来决定，输的人就去打扫厕所。

游戏规则为：阿明和阿强分别掷一个骰子，每个骰子的点数为 1 到 6，第 1 轮掷骰子时，若点数和为 2 或 3，阿明获胜，输出 1 win；第 2 轮掷骰子时，若点数和为 11 或 12，阿强获胜，输出 2 win；第 3 轮掷骰子时，若点数和为 2 或 3，阿明获胜，输出 1 win；第 4 轮掷骰子时，若点数和为 11 或 12，阿强获胜，输出 2 win...

以此规律类推，直到有人胜利则游戏结束。若每 1 轮中没有获胜的人，则继续下一轮游戏。过程中，输出每一轮二人骰子点数之和。

【输出样例 1】

```
round 1: 9
round 2: 11
2 win
```

【输出样例 2】

```
round 1: 7
round 2: 4
round 3: 3
1 win
```

(注意输出格式，要求英文冒号，看清空格位置)

参考代码：

```
import random
cnt = 0
while True:
    cnt += 1
    ming = random.randint(1, 6)
    qiang = random.randint(1, 6)
    print('round ', cnt, ': ', ming + qiang, sep='')
    if cnt % 2 == 1 and ming + qiang in [2, 3]:
        print('1 win')
        break
    elif cnt % 2 == 0 and ming + qiang in [11, 12]:
        print('2 win')
        break
```

#### 5、小 Y 的成绩

小 Y 特别喜欢环湖赛跑，因为这样空气非常好，跑起来特别有动力。正值一年一度的环湖跑王者争霸赛开始了，小 Y 就立刻报名了。比赛主办方在周长

为 8000 米的湖周围一圈设置了  $n$  个夺分点，逆时针编号为  $1\sim n$ ，每个夺分点的积分不一定相同，每个积分点的具体位置和积分数量由主办方在比赛开始前公布。比赛规定：参赛者可以选择从赛道的任意一点出发，只能逆时针跑，需要在 5 分钟内尽可能多拿分，只要到达积分点即可得到该积分点的积分，不同人到达同一积分点均能得到积分，互不影响。经过小 Y 自己赛前的反复测试发现，平均下来，自己比赛时的速度为 3.5 米每秒。请问小 Y 以这个速度最多能拿多少分呢？最多能拿的分和最少能拿的分差多少呢？

**【输入格式】**

$2n + 1$  行，

第一行，一个整数  $n$ ，表示积分点的个数；

接下来  $n$  行，每行一个数，表示积分点的距离，这个距离是相对于 1 号积分点的距离。距离按照积分点的编号从 1 到  $n$  进行输入，由小到大，保证一圈的距离足够大，无法跑完一整圈；

再接下来的  $n$  行，每行一个数，表示每个积分点的积分，也是按照积分点的编号顺序进行输入，保证积分总和不超过 1000000000。

**【输出格式】**

两行，

第一行为小 Y 能得到的最多积分；

第二行为小 Y 能拿到最多积分与最少积分的差值。

**【输入样例】**

```
5
0
123
555
6111
7521
23
54
21
76
32
```

**【输出样例】**

```
130
109
```

**参考代码：**

```
n = int(input())
dis = []
for i in range(n):
    d = int(input())
    dis.append(d)
scores = []
for i in range(n):
    s = int(input())
```

```

        scores.append(s)
maxn = -1
minn = 1e9
# 枚举所有可能的起点，起点一定是某个积分点之一
for i in range(0, n):
    y_dis = 5 * 60 * 3.5
    tot = scores[i]
    cur = (i + 1) % n
    while y_dis >= (dis[cur] + 8000 - dis[cur - 1]) %
8000:
        y_dis -= (dis[cur] + 8000 - dis[cur - 1]) %
8000
        tot += scores[cur]
        cur = (cur + 1) % n
    maxn = max(maxn, tot)
    minn = min(minn, tot)
print(maxn)
print(maxn - minn)

```

## 6、回家的路

小 Z 家所在的街道都是横平竖直的，从空中俯视看，非常像一张方格表，由  $n$  行  $m$  列的方格组成，而小 Z 此刻正站在最左上角的格子中，想走回到最右下角的家中，他每次只能往右或者往下走一个格子，毕竟不能走回头路。由于小 Z 家附近在修路，就导致有些格子还不能走。好在小 Z 手上有一份地图，标注了哪些格子能走，哪些格子不能走。现在请你帮小 Z 算算他这次回家一共有多少种走法吧~

### 【输入格式】

共  $n + 1$  行，

第 1 行为 2 个正整数  $n$ 、 $m$ ，用空格隔开，表示方格表的行数和列数；

第 2 ~  $n+1$  行为地图，每行为  $m-1$  个用空格隔开的正整数 0 或 1，0 表示不能走，1 表示能走。

### 【输出格式】

一行，一个数，表示小 Z 回家可选的路线总数。

### 【输入样例】

```

3 4
1 0 1 1

```



```
1 1 1 1
```

```
1 1 1 1
```

【输出样例】

```
4
```

参考代码:

```
n, m = [int(i) for i in input().split(' ')]
```

```
ls = [[0] * (m + 1)]
```

```
for i in range(n):
```

```
    ls.append([0] + [int(i) for i in input().split(' ')])
```

```
for i in range(1, n + 1):
```

```
    for j in range(1, m + 1):
```

```
        if ls[i][j] == 0 or i == 1 and j == 1:
```

```
            continue
```

```
        ls[i][j] = ls[i - 1][j] + ls[i][j - 1]
```

```
print(ls[n][m])
```