



# Python 三级

2023 年 9 月

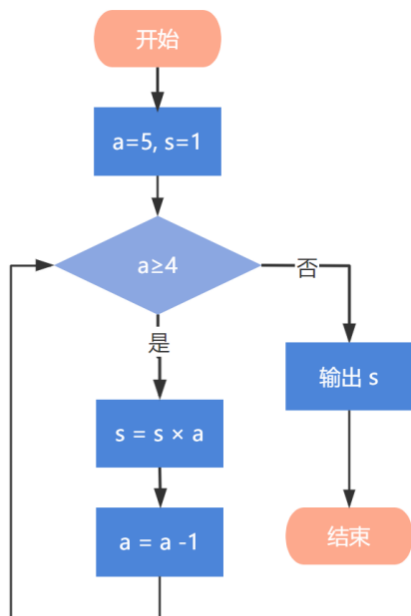
## 1 单选题（每题 2 分，共 30 分）

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	B	B	C	D	D	D	C	A	A	B	C	B	B	B	C

第 1 题 人们所使用的手机上安装的App通常指的是（ ）。

- ☐ A. 一款操作系统
- ☐ B. 一款应用软件
- ☐ C. 一种通话设备
- ☐ D. 以上都不对

第 2 题 下列流程图的输出结果是（ ）。



- ☐ A. 60
- ☐ B. 20
- ☐ C. 5
- ☐ D. 1

第3题 已知大写字符 'A' 的ASCII编码的十六进制表示为 0x41，则字符 'L' 的ASCII编码的十六进制表示为（ ）。

- ☐ A. 4A
- ☐ B. 4B
- ☐ C. 4C
- ☐ D. 52

第4题 下面有关Python的dict类型key（键）的说法，正确的是（ ）。

- ☐ A. dict类型的key可以是list数据
- ☐ B. dict类型的key可以是set数据
- ☐ C. dict类型的key可以是dict数据
- ☐ D. dict类型的key可以是tuple数据

第5题 下面Python赋值语句，不正确的是（ ）。

- ☐ A. a = [12, 12]
- ☐ B. b = (12, 12)
- ☐ C. c = {12, 12}
- ☐ D. 以上没有不正确

第6题 有关下面Python代码的描述，正确的是（ ）。

```
a = (1, 2, 3, 4)
b = (5, 6, 7, 8)
a = a[1:] + b[:1]
print(a)
```

- ☐ A. 上述代码执行将报错
- ☐ B. 加号运算符不适用tuple类型
- ☐ C. tuple是不可更改类型，不能切片（slice）操作
- ☐ D. 输出结果是(2, 3, 4, 5)

第7题 下列Python赋值语句错误的是（ ）。

- ☐ A. a = [1, 2] + [3, 4]
- ☐ B. b = (1, 2) + (3, 4)
- ☐ C. c = {1, 2} + {3, 4}
- ☐ D. d = "1, 2" + "3, 4"

第8题 下面Python代码执行后的输出是（ ）。

```
lstA = [11, 12, 13, 14, 15, 16, 17]
print(lstA[3: 2], end = ",")
print(lstA[-5: -3])
```

- ☐ A. [], [13, 14]
- ☐ B. [13, 14], []
- ☐ C. [13, 14], [13, 14]
- ☐ D. [], []

第9题 下面Python代码执行后的输出是（ ）。

```
strA = "A1x"
strA = sorted(strA)
print(strA)
```

- ☐ A. ['1', 'A', 'x']
- ☐ B. ['1', 'x', 'A']
- ☐ C. "1Ax"
- ☐ D. "A1x"

第10题 下面Python代码执行后的输出是（ ）。

```
lstA = [2, -3, 4, 3, 5, 6, 7, 8, -2]
lstA.sort(key = abs, reverse = True)
print(lstA)
```

- ☐ A. [8, 7, 6, 5, 4, -3, 3, -2, 2]
- ☐ B. [8, 7, 6, 5, 4, -3, 3, 2, -2]
- ☐ C. [2, -2, -3, 3, 4, 5, 6, 7, 8]
- ☐ D. [-2, 2, -3, 3, 4, 5, 6, 7, 8]

第11题 下面Python代码执行后的输出是（ ）。

```
setA = set("Python")
setB = set("python")
print(setA - setB)
```

- ☐ A. 'P'

- ☐ B. 'p'
- ☐ C. {'P'}
- ☐ D. {'p'}

**第12题** 某机构职工编码规则是“入职年份顺序码”，其中入职年份为2位数，顺序码3位数表示入职顺序。下面代码要求输入职工编码，并在输入大写字母Q时结束输入，要求输出每个年份入职人数，应在横线填入代码是（ ）。

```
statData = {} #保存每个年份对应的人数
while True:
    ID = input("请输入职工编号: ")
    if ID == "Q":
        break
    Year = ID[:2] #得到入职年份
    _____
print(statData)
```

- ☐ A. statData[Year] += statData.get(Year, 0) + 1
- ☐ B. statData[Year] = statData.get(Year, 0) + 1
- ☐ C. statData[Year] = statData.get(Year, None) + 1
- ☐ D. statData[Year] += 1

**第13题** Python表达式 [x for x in range(2,20) if x % 3 and x % 5 == 0] 的值是（ ）。

- ☐ A. [15]
- ☐ B. [5, 10]
- ☐ C. [2, 4, 7, 8, 11, 13, 14, 16, 17, 19]
- ☐ D. [3, 6, 9, 12, 18]

**第14题** 下面Python代码用于实现十进制到M进制（M为二进制到十六进制）的转换，横线处应填入代码是（ ）。

```
N = int(input("请输入正整数: "))
M = int(input("请输入转换进制, 不能大于
16: "))
Alpha="0123456789ABCDEF"

rst = "" #保存转换结果
while N != 0:
    Remainder = N % M #Remainder表示余
    数
    _____
    N = N // M
print(rst)
```

- ☐ A. `rst += Alpha[Remainder]`
- ☐ B. `rst = Alpha[Remainder] + rst`
- ☐ C. `rst = rst + Alpha[Remainder]`
- ☐ D. `rst = str(Remainder) + rst`

第15题 下面Python代码实现字符串乱序【随机打乱】，横线处应填入代码（ ）。

```
import random
a = "123ABCxyz"
a = list(a)
a.sort(_____)
print("".join(a))
```

- ☐ A. `key = random.random`
- ☐ B. `key = lambda x:random.random(x)`
- ☐ C. `key = lambda x:random.random()`
- ☐ D. `key = lambda x:random.random`

## 2 判断题（每题2分，共20分）

题号	1	2	3	4	5	6	7	8	9	10
答案	√	×	×	×	×	√	×	√	×	×

第1题 我们常说的互联网（Internet）是一个覆盖全球的广域网络，它不属于任何一个国家。

第2题 二进制数 101.101 在十进制下是 5.005。

第3题 Python内置函数len()对list、tuple和str有效，对set和dict无效。

第4题 Python表达式len(set("12,34,56"))的值为3。

第5题

Python表达式2&3的值为5。

第6题 执行Python代码print("李白".replace("白","杜"))后将输出"李杜"。

第7题 执行Python代码print(list(filter(lambda n:n%5,range(10))))后将输出[0, 5]。

第8题 执行Python代码print([i for i in range(10) if i % 5 == 0])后将输出[0, 5]。

第9题 下面Python代码执行时输入10,20后将输出30。

```
a,b = input().split(",")
print(a + b)
```

第10题 在Python代码中先执行 `a = (5,6,3,10)`，然后执行`print(a.sort())`其输出结果是None。

### 3 编程题（每题 25 分，共 50 分）

#### 3.1 编程题 1

- 试题编号：2023-09-23-03-P-01
- 试题名称：小杨的储蓄
- 时间限制：1.0 s
- 内存限制：128.0 MB

##### 3.1.1 问题描述

小杨共有  $N$  个储蓄罐，编号从 0 到  $N - 1$ 。从第 1 天开始，小杨每天都会往存钱罐里存钱。具体来说，第  $i$  天他会挑选一个存钱罐  $a_i$ ，并存入  $i$  元钱。过了  $D$  天后，他已经忘记每个储蓄罐里都存了多少钱了，你能帮帮他吗？

##### 3.1.2 输入描述

输入 2 行，第一行两个整数  $N, D$ ；第二行  $D$  个整数，其中第  $i$  个整数为  $a_i$ （保证  $0 \leq a_i \leq N - 1$ ）。

每行的各个整数之间用单个空格分隔。

保证  $1 \leq N \leq 1,000$ ； $1 \leq D \leq 1,000$

##### 3.1.3 输出描述

输出  $N$  个用单个空格隔开的整数，其中第  $i$  个整数表示编号为  $i - 1$  的存钱罐中有多少钱（ $i = 1, \dots, N$ ）。

##### 3.1.4 特别提醒

在常规程序中，输入、输出时提供提示是好习惯。但在本场考试中，由于系统限定，请不要在输入、输出中附带任何提示信息。

##### 3.1.5 样例输入 1

```
1 | 2 3
2 | 0 1 0
```

##### 3.1.6 样例输出 1

```
1 | 4 2
```

##### 3.1.7 样例解释 1

小杨在第 1 天、第 2 天、第 3 天分别向 0 号、1 号、0 号存钱罐存了 1 元钱、2 元钱、3 元钱，因此 0 号存钱罐有  $1 + 3 = 4$  元钱，而 1 号存钱罐有 2 元钱。

##### 3.1.8 样例输入 2

```
1 | 3 5
2 | 0 0 0 2 0
```

### 3.1.9 样例输出 2

```
1 11 0 4
```

### 3.1.10 参考程序

```
1 n, d = map(int, input().split(' '))
2 x = list(map(int, input().split(' ')))
3 arr = [0 for i in range(n)]
4 for i in range(d):
5     arr[x[i]] += (i+1)
6 print(' '.join(map(str, arr)))
```

## 3.2 编程题 2

- 试题名称：进制判断
- 时间限制：1.0 s
- 内存限制：128.0 MB

### 3.2.1 问题描述

$N$  进制数指的是逢  $N$  进一的计数制。例如，人们日常生活中大多使用十进制计数，而计算机底层则一般使用二进制。除此之外，八进制和十六进制在一些场合也是常用的计数制（十六进制中，一般使用字母 A 至 F 表示十至十五）。

现在有  $N$  个数，请你分别判断他们是否可能是二进制、八进制、十进制、十六进制。例如，15A6F 就只可能是十六进制，而 1011 则是四种进制皆有可能。

### 3.2.2 输入描述

输入的第一行为一个十进制表示的整数  $N$ 。接下来  $N$  行，每行一个字符串，表示需要判断的数。保证所有字符串均由数字和大写字母组成，且不以 0 开头。保证不会出现空行。

保证  $1 \leq N \leq 1000$ ，保证所有字符串长度不超过 10。

### 3.2.3 输出描述

输出  $N$  行，每行 4 个数，用空格隔开，分别表示给定的字符串是否可能表示一个二进制数、八进制数、十进制数、十六进制数。使用 1 表示可能，使用 0 表示不可能。

例如，对于只可能是十六进制数的 15A6F，就需要输出 0 0 0 1；而对于四者皆有可能的 1011，则需要输出 1 1 1 1。

### 3.2.4 特别提醒

在常规程序中，输入、输出时提供提示是好习惯。但在本场考试中，由于系统限定，请不要在输入、输出中附带任何提示信息。

### 3.2.5 样例输入 1

```
1 2
2 15A6F
3 1011
```

### 3.2.6 样例输出 1

```
1 0 0 0 1
2 1 1 1 1
```

### 3.2.7 样例输入 2

```
1 4
2 1234567
3 12345678
4 FF
5 GG
```

### 3.2.8 样例输出 2

```
1 0 1 1 1
2 0 0 1 1
3 0 0 0 1
4 0 0 0 0
```

### 3.2.9 参考程序

```
1 n = int(input())
2 for i in range(n):
3     st = input().strip()
4     max_bit = '0'
5     for ch in st:
6         if ch > max_bit:
7             max_bit = ch
8     answer = [
9         max_bit <= '1',
10        max_bit <= '7',
11        max_bit <= '9',
12        max_bit <= 'F',
13    ]
14    print(' '.join(map(str, map(int, answer))))
```