



Python 四级

2026 年 03 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	D	C	C	B	C	B	D	C	B	C	B	A	B	D

第 1 题 2026年春节联欢晚会上一个武术表演节目《武 BOT》。节目中多个人形机器人会表演空翻，它们落地可能会有微微踉跄，但都会迅速调整姿态站稳，并适当移动来和前后左右的其他机器人保持原来队列。如果将机器人视为一个计算机系统，那么在该计算机系统中下面哪一项不能作为输入设备()。

- A. 用于检测重心的重力传感器
- B. 预装的AI算法程序
- C. 接收动作指令的遥控器
- D. 拍摄其他机器人的摄像头

第 2 题 下面代码用来找出输入的 N 个正整数中最大的一个。如果将代码段用流程图来表示，则 L1 标记的代码行应该使用的图形是 ()。

```
1 Max = 0
2 N = int(input())
3
4 while(N):
5     val = int(input())
6     if val > Max: #L1
7         Max = val
8     N -= 1
9
10 print(Max)
```

- A. 圆形框
- B. 椭圆形框
- C. 平行四边形框
- D. 菱形框

第 3 题 执行以下 Python 代码后，输出的结果是? ()

```
1 matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
2 total = 0
3 for row in matrix:
4     for element in row:
5         if element % 2 == 0:
6             total += element
7 print(total)
```

- A. 4
- B. 10
- C. 20
- D. 25

第4题 执行以下Python代码，说法正确的是？（）

```
1 | tup = (1, [2, 3], 4)
2 | tup[1] = [5, 6]
```

- A. 元组被成功修改为 (1, [5, 6], 4)
- B. 代码可以执行，但元组内容未改变
- C. 抛出 `TypeError` 异常
- D. 元组被修改为 (1, 2, 3, 4, 5, 6)

第5题 执行以下Python代码后，输出的结果是？（）

```
1 | words = ["hello", "world", "python"]
2 | result = [char for word in words for char in word if char in 'aeiou']
3 | print(len(result))
```

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

第6题 函数定义 `def func(*args): pass` 中的 `*args` 会将传入的参数打包成什么类型？

- A. 列表
- B. 字典
- C. 元组
- D. 集合

第7题 在嵌套函数中，需要修改外层函数（非全局作用域）的变量，应使用哪个关键字？（）

- A. `global`
- B. `nonlocal`
- C. `local`
- D. `outer`

第8题 执行以下Python代码后，输出的结果是？（）

```

1 def process_data(data):
2     result = []
3     for item in data:
4         if item > 0:
5             result.append(item * 2)
6         elif item < 0:
7             result.append(abs(item))
8         else:
9             result.append(0)
10    return result
11
12
13 output = process_data([-2, 0, 3, -1])
14 print(sum(output))

```

- A. 12
- B. 6
- C. 8
- D. 9

第9题 关于 lambda 匿名函数，以下描述正确的是？（）

- A. lambda 匿名函数可以包含多个表达式，用分号分隔
- B. lambda 匿名函数必须有参数
- C. lambda 匿名函数可以接受多个参数，但只能有一个表达式
- D. lambda 匿名函数必须有 return 语句

第10题 文件 text.txt 的内容如下：

```

1 Python is a programming language.
2 Python is easy to learn.

```

以下程序统计单词 "Python" 在文件中出现的次数。Python 后面是空格或行尾，且不跟标点。请补全代码。（）

```

1 count = 0
2 with open("text.txt", "r") as f:
3     content = f._____ # 补全代码
4     words = content.split()
5     for word in words:
6         if word == "Python":
7             count += 1
8 print("Python出现次数: ", count)

```

- A. readlines()
- B. read()
- C. readline()
- D. readlines

第11题 以下代码的时间复杂度是？（）

```

1 def func(n):
2     sum = 0
3     for i in range(n):
4         for j in range(i):
5             sum += j

```

- A. $O(1)$
- B. $O(n)$
- C. $O(n^2)$
- D. $O(\log n)$

第 12 题 执行以下 Python 代码后，输出的结果是？（）

```

1 try:
2     print(1, end="")
3     raise ValueError("x")
4 except ValueError as e:
5     print(2, end="")
6 else:
7     print(3, end="")
8 finally:
9     print(4, end="")

```

- A. 14
- B. 124
- C. 1234
- D. 程序报错

第 13 题 以下代码使用插入排序对列表进行升序排序。空白处应填入？（）

```

1 def insertion_sort_optimized(arr):
2     n = len(arr)
3     for i in range(1, n):
4         key = arr[i] # 当前待插入元素
5         j = i - 1
6         while j >= 0 and arr[j] > key:
7             arr[j + 1] = arr[j] # 元素后移
8             j -= 1
9         _____ # 插入元素到正确位置

```

- A. `arr[j + 1] = key`
- B. `arr[j - 1] = key`
- C. `key = arr[j]`
- D. `arr[i] = key`

第 14 题 阅读以下 Python 代码，下列说法正确的是？（）

```

1 n = int(input())
2
3 def is_palindrome(s):
4     return s == s[::-1]
5
6 max_palindrome = 0
7 for i in range(10 ** (n - 1), 10 ** n):
8     for j in range(i, 10 ** n):
9         product = i * j
10        if is_palindrome(str(product)) and product > max_palindrome:
11            max_palindrome = product
12 print(max_palindrome)

```

- A. 找出所有 n 位数

- B. 找出由两个 n 位数相乘得到的最大回文数
- C. 找出所有回文数
- D. 计算两个 n 位数乘积的最大值

第 15 题 以下代码实现了根据分数返回等级 (A: 90-100, B: 80-89, C: 70-79, D: 60-69, F: 0-59) 的功能, 请选择正确的选项填入①②处。 ()

```

1 | grade_map = {90: "A", 80: "B", 70: "C", 60: "D", 0: "F"}
2 |
3 | def get_grade(score):
4 |     if score < 0 or score > 100:
5 |         return "无效分数"
6 |     for key in sorted(grade_map.keys(), reverse=True):
7 |         if score >= key:
8 |             return _____①_____
9 |
10 | score = int(input())
11 | grade = get_grade(score)
12 | print(f"分数 {score} 的等级是 {_____②_____}")

```

- A. ① grade_map[score] ② grade
- B. ① key ② get_grade(score)
- C. ① grade_map[key] ② score
- D. ① grade_map[key] ② grade

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 小明的妈妈最近刚刚给他买了一块电话手表, 除了可以看时间, 小明也可以用它和妈妈打电话、收发信息, 那么可以推测这块手表中装有一款特定操作系统。 ()

第 2 题 在 Python 语言中, 一个集合可以作为另一个集合的元素。

第 3 题 执行以下 Python 代码后, 输出的结果是 <class 'set'>。

```

1 | my_set = {i: j for i, j in zip([1, 2], [3, 4])}
2 | print(type(my_set))

```

第 4 题 执行以下 Python 代码后, 输出的结果是 20 10。

```

1 | def func(a, b):
2 |     a = a ^ b
3 |     b = a ^ b
4 |     a = a ^ b
5 |     return a, b
6 |
7 |
8 | a, b = func(10, 20)
9 | print(a, b)

```

第 5 题 执行以下 Python 代码后, 输出的结果是 [6, 7]。

```

1 | print(list(filter(lambda x: x > 5, [3, 6, 7, 2])))

```

第 6 题 执行以下 Python 代码后，输出的结果是 '789\n456'。

```
1 with open('test.txt', 'w') as f:
2     f.write('123\n456')
3 with open('test.txt', 'r+') as f:
4     f.seek(0, 2)
5     f.write('789')
6     f.seek(0)
7     print(repr(f.read()))
```

第 7 题 下面这段程序的时间复杂度为平方阶 $O(n)$ 。

```
1 def func(n):
2     for i in range(n):
3         for j in range(n):
4             if i % 2 == 0:
5                 break
```

第 8 题 执行以下 Python 代码后，会抛出异常 `ZeroDivisionError`。

```
1 data = {'a': 1}
2 index = int(input())
3 print(data['b'] / index)
```

第 9 题 在对列表进行排序时，`sorted()` 函数会返回一个新的排序后的列表，不修改原列表，而 `sort()` 方法会直接修改原列表，返回值为 `None`。

第 10 题 阅读以下递推算法的 Python 代码，执行后的输出结果是 13。

```
1 a, b = 1, 2
2 for i in range(3):
3     a, b = b, a + b
4 print(b)
```

3 编程题（每题 25 分，共 50 分）

3.1 编程题 1

- 试题名称：山之谷
- 时间限制：1.0 s
- 内存限制：512.0 MB

3.1.1 题目描述

现有一片山地，可以视为一个 N 行 M 列的网格图，第 i 行 j 列的海拔为 $h_{i,j}$ 。

如果一个单元格的海拔不高于其所有相邻单元格（相邻包括上、下、左、右、左上、右上、左下、右下，最多 8 个方向）的海拔，则称该单元格为山谷。

请你数一数该片中有多少山谷。

3.1.2 输入格式

第一行包含 2 个整数 N, M ，表示山地的大小。

之后 N 行，每行包含 M 个整数 $h_{i,1}, h_{i,2}, \dots, h_{i,M}$ ，表示海拔。

3.1.3 输出格式

输出 1 行，包含 1 个整数 C ，表示山谷的数量。

3.1.4 样例

3.1.4.1 输入样例

```
1 | 3 5
2 | 7 6 6 7 9
3 | 6 5 6 7 6
4 | 6 5 7 8 9
```

3.1.4.2 输出样例

```
1 | 3
```

3.1.5 样例解释

样例 1 如图所示，绿色单元格代表山谷：

7	6	6	7	9
6	5	6	7	6
6	5	7	8	9

3.1.6 数据范围

保证 $1 \leq N, M \leq 100$ ， $1 \leq h_{i,j} \leq 10^5$ 。

3.1.7 参考程序

```
1 # 输入行列数
2 n, m = map(int, input().split())
3
4 # 输入每一行的海拔
5 h = []
6 for i in range(n):
7     row = list(map(int, input().split()))
8     h.append(row)
9
10 ans = 0
11 # 先遍历行
12 for i in range(n):
13     # 再遍历列
14     for j in range(m):
15         ok = True
16         # 遍历周边8个格子
17         for k in range(-1, 2):
18             for l in range(-1, 2):
19                 # 边界判定, 不能超列表边界
20                 if 0 <= i + k < n and 0 <= j + l < m and h[i + k][j + l] < h[i][j]:
21                     ok = False
22             if(ok):
23                 ans += 1
24 print(ans)
```

3.2 编程题 2

- 试题名称: 礼盒排序
- 时间限制: 1.0 s
- 内存限制: 512.0 MB

3.2.1 题目描述

商店推出了许多礼盒, 每个礼盒中包含 k 件商品, 每件商品都有一个价格。

现在需要对这些礼盒进行排序, 排序规则如下:

1. 先按礼盒总价格从小到大排序;
2. 如果总价格相同, 按礼盒中最贵商品的价格从小到大排序;
3. 如果仍然相同, 按礼盒中最便宜商品的价格从小到大排序;
4. 如果仍然相同, 按礼盒编号从小到大排序。

请输出排序后的礼盒编号。

3.2.2 输入格式

第一行包含两个整数 n 和 k , 分别表示礼盒数量和每个礼盒中商品的数量。

接下来 n 行, 每行包含 k 个整数, 第 i 行表示第 i 个礼盒中各商品的价格。

3.2.3 输出格式

输出一行，包含排序后的礼盒编号（编号从 1 开始），用空格分隔。

3.2.4 样例

3.2.4.1 输入样例

```
1 | 4 3
2 | 3 5 2
3 | 4 1 5
4 | 2 2 4
5 | 3 4 3
```

3.2.4.2 输出样例

```
1 | 3 4 2 1
```

3.2.5 样例解释

4 个礼盒分别为：

编号	商品价格	总价	最大值	最小值
1	3 5 2	10	5	2
2	4 1 5	10	5	1
3	2 2 4	8	4	2
4	3 4 3	10	4	3

排序过程：

1. 按总价排序，3号礼盒总价最小；
2. 其余总价均为 10，再按最大值排序，4号最大值更小；
3. 1号和2号最大值相同，再按最小值排序，2号更小。

最终顺序为：3 4 2 1

3.2.6 数据范围

保证 $1 \leq n \leq 10^3$ ， $1 \leq k \leq 10$ ，商品价格 $\leq 10^4$ 。

3.2.7 参考程序

```
1 | n, k = map(int, input().split())
2 |
3 | v = []
4 |
5 | for i in range(n):
6 |     nums = list(map(int, input().split()))
7 |     s = sum(nums)
8 |     mx = max(nums)
9 |     mn = min(nums)
10 |    v.append((s, mx, mn, i + 1))
11 |
12 | v.sort()
13 |
14 | for i in range(n):
15 |     print(v[i][3], end=" " if i == n - 1 else " ")
16 | print()
```