

青少年软件编程 (Python) 等级考试试卷 (五级)

分数：100 题数：38

一、单选题(共 25 题，共 50 分)

1. 天文学社的小明需要用 Python 程序正确输出地月平均距离 (384400 公里) 并转换为万公里单位。以下代码的输出结果是什么? ()

```
distance = 384400  
print("地月平均距离：{:.1f}万公里".format(distance / 10000))
```

- A. 地月平均距离：384400.0 万公里
- B. 地月平均距离：38.4 万公里
- C. 地月平均距离：38.44 万公里
- D. TypeError: 不支持整数除法

试题编号：2025-zpf-009

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：原始地月距离为 384400 公里，distance / 10000 将其转换为 38.44 万公里，{:.1f}保留 1 位小数，四舍五入后得到 38.4 万公里。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

2. 某基因分析程序需截取 DNA 序列 ("ATCGTA") 的第 2 到第 4 个碱基并大写输出。以下代码的输出结果是什么? ()

```
dna = "ATCGTA"  
print(dna[1:4].upper())
```

- A. TCG
- B. TCGT
- C. atc
- D. IndexError: 索引超出范围

试题编号：2025-zpf-010

试题类型：单选题

标准答案：A

试题难度：一般

试题解析：dna[1:4]截取索引 1 到 3 的字符 ("TCG")，upper()将其转为大写 (原序列已大写)。

考生答案：A

考生得分：2

是否评分：已评分

评价描述：

3. 小明在 Python 中输入了一个单词 word = "elephant"，他想找到这个单词的第 4 个字母。以下哪段代码能正确实现? ()

- A. word[4]
- B. word[0:4]
- C. word[3]
- D. word[0:3]

试题编号：2025-zpf-011

试题类型：单选题

标准答案：C

试题难度：容易

试题解析：在 Python 中，使用方括号[]进行索引访问，第 4 个字母是 'p'，对应的索引是 3，故选项 C 正确。

考生答案：C

考生得分：2

是否评分：已评分

评价描述：

4. 小红需要将变量 `score = 95.5` 格式化为以下输出：

"本次数学考试得分：95.50"

她应该使用的代码是？（ ）

- A. `print(f"本次数学考试得分：{score}")`
- B. `print("本次数学考试得分：%.2f" % score)`
- C. `print("本次数学考试得分：{score:.1f}")`
- D. `print("本次数学考试得分：" + str(score))`

试题编号：2025-zpf-012

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：只有 B 选项通过 `%.2f` 严格保证输出两位小数（95.50）。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

5. 小明的学习小组需要整理数学考试成绩，将高于等于 80 分的记录标记为 "A"，60-79 分标记为 "B"，其余标记为 "C"。已知成绩列表为 `scores = [75, 92, 58, 84, 65]`，以下哪种推导式能正确生成等级列表？（ ）

- A. `["A" for x in scores if x >=80] + ["B" for x in scores if 60<=x<80] + ["C" for x in scores if x<60]`
- B. `["A" if x >= 80 else "B" if x >= 60 else "C" for x in scores]`
- C. `[x >=80 ? "A" : x >=60 ? "B" : "C" for x in scores]`
- D. `{ "A" if x >=80 else "B" if x >=60 else "C" for x in scores }`

试题编号：2025-zpf-027

试题类型：单选题

标准答案：B

试题难度：较难

试题解析：A 选项错误地将原列表拆分处理导致顺序丢失，C 使用无效的三目运算符语法，D 生成的是集合而非列表。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

6. 已知选修课学生名单 `course1 = {"张三", "李四", "王五"}`, `course2 = {"李四", "赵六", "王五"}`。要找出同时选修两门课的学生，应使用的推导式是？（ ）

- A. `{x for x in course1 if x in course2}`
- B. `[x for x in course1 and course2]`
- C. `{x for x in course1 or course2}`
- D. `{x for x in course1 if x not in course2}`

试题编号：2025-zpf-028

试题类型：单选题

标准答案：A

试题难度：较难

试题解析：选项 B 语法错误（列表+逻辑运算符无效），选项 C 求并集，选项 D 求差集。

选项 A 通过遍历 `course1` 并检查存在性得到正确交集。

考生答案：A

考生得分：2

是否评分：已评分

评价描述：

7. 班级组织户外活动，老师要统计带水杯的同学名单。已知学生信息列表为：

`students = [{"name": "小明", "has_cup": True}, {"name": "小红", "has_cup": False}, {"name": "小刚", "has_cup": True}]`，正确的推导式是？（ ）

- A. `[student["name"] for student in students if has_cup]`
- B. `[name for student in students where student.has_cup]`
- C. `[student["name"] for student in students if student["has_cup"]]`
- D. `[student.name for student in students when student.has_cup]`

试题编号：2025-zpf-029

试题类型：单选题

标准答案：C

试题难度：较难

试题解析：A 选项的 `has_cup` 未指明属于 `student` 字典；B 和 D 语法错误（Python 无 `where` 和 `when` 关键字）。

考生答案：C

考生得分：2

是否评分：已评分

评价描述：

8. 下列 Python 代码的输出结果是？（ ）

```
for i in range(5, 17, 3):  
    print(i, end=',')
```

- A. 5,8,11,14
- B. 5,8,11,14,
- C. 5,8,11,14,17
- D. 5,8,11,14,17,

试题编号：20250420-lfy-015

试题类型：单选题

标准答案：B

试题难度：容易

试题解析：range(5, 17, 3)生成的整数序列是[5,8,11,14]，不包含最终值 17，且输出时 print()函数指定了结尾符为“，”，需要以逗号结尾，故结果为 5,8,11,14,。

考生答案：A

考生得分：0

是否评分：已评分

评价描述：

9. 以下 Python 程序段需要 print(i)语句执行 5 次，则①处不能填入的是？（ ）

```
for i in range( ① ):
    print(i)
```

- A. 5
- B. 1,11,2
- C. -2,-7
- D. -1,-14,-3

试题编号：20250420-lfy-016

试题类型：单选题

标准答案：C

试题难度：容易

试题解析：range(start, stop, step) 函数包含 3 个参数，其中 start 省略，默认从 0 开始，step 省略，默认为 1。选项中 range(-2,-7)，因为 $-2 > -7$ ，而步长为 1， $-2+1 > -7$ ，无法生成序列。

考生答案：D

考生得分：0

是否评分：已评分

评价描述：

10. 韩信带 1500 名士兵打仗，战死四五百人，剩下的士兵排队：站 3 人一排，多出 2 人；站 5 人一排，多出 4 人；站 7 人一排，多出 6 人。下列代码中不能正确计算剩余士兵人数的是？（ ）

- A.

```
for x in range(998,1100,3):
    if x%5==4 and x%7==6:
        print(x)
```
- B.

```
for x in range(1100,1000,-3):
    if x%5==4 and x%7==6:
        print(x)
```
- C.

```
for x in range(1000,1100,3):
```

```
    if x%5==4 and x%7==6 :
        print(x)
D. for x in range(1000,1100,1) :
    if x%3==2 and x%5==4 and x%7==6 :
        print(x)
```

试题编号：20250420-lfy-017

试题类型：单选题

标准答案：C

试题难度：一般

试题解析：range(998,1100,3)和 range(1100,1000,-3)的起始值已经符合除 3 取余为 2，再加上步长为 3，循环变量 i 始终满足题干条件站 3 人一排多出 2 人。而 range(1000,1100,3)中起始值 1000 不符合除 3 取余为 2，且 if 条件中没有 x%3==2 语句进行判断，故不能正确计算。

考生答案：D

考生得分：0

是否评分：已评分

评价描述：

11. 某 Python 程序如下，运行后输出的图形是？（ ）

```
s=""
for i in range(3):
    for j in range(1,i+2,2):
        s=s+'*'
    print(s)
```

- A. *
- **
- ****
- B. ***
- ***
- ***
- C. ***
- **
- *
- D. **
- ****
- *****

试题编号：20250420-lfy-018

试题类型：单选题

标准答案：A

试题难度：容易

试题解析：

外层循环执行 1 次，内层循环执行 1 轮。

i=0 时，j=1，内层循环执行 1 次，输出 1 颗*

i=1 时，j 的范围是[1]，内层循环执行 1 次，共输出 2 颗*

i=2 时，j 的范围是[1,3]，内层循环执行 2 次，共输出 4 颗*

考生答案：A

考生得分：2

是否评分：已评分

评价描述：

12.某 Python 程序如下：

```
import random
list1=[1,2,3,4,5,6,7,8,9,10,11,12]
s=random._____(list1,6)
print(s)
```

现想随机取出列表 list1 中的 6 个元素，划线处应填入的代码为？（ ）

A. uniform

B. choice

C. shuffle

D. sample

试题编号：20250420-lfy-031

试题类型：单选题

标准答案：D

试题难度：容易

试题解析：

random.uniform(a,b) 表示随机生成一个 [a,b] 范围内的实数，不符合题意；

random.choice(seq) 表示从序列 seq 的元素中随机挑选一个元素，不符合题意；

random.shuffle(seq) 表示将序列 seq 的所有元素随机排序，不符合题意；

random.sample(seq,k) 表示从序列 seq 中随机挑选 k 个元素，符合题意。

考生答案：B

考生得分：0

是否评分：已评分

评价描述：

13. 某 Python 程序如下：

```
import math
x=math.ceil(5/3)
y=math.floor(11/3)
if x>y:
    print(math.pow(y,2))
else:
    print(math.pow(x,2))
```

程序运行后，输出的结果是？（ ）

A. 1.0

- B. 4.0
- C. 9.0
- D. 16.0

试题编号：20250420-lfy-032

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：x=math.ceil(5/3)，向上取整结果为 2，y=math.floor(11/3)，向下取整结果为 3，对 x 和 y 中的较小值进行平方运算，最终 math.pow(2,2)结果为 4.0。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

14. 在 Python 中，利用 datetime 模块的 strftime()方法，可以将日期和时间格式化成字符串。下列错误的格式是？（ ）

- A. now.strftime("%y/%m/%d")
- B. now.strftime("%y-%m-%d")
- C. now.strftime("%H:%M:%S")
- D. now.strftime("%H 时%M 分%S 秒")

试题编号：20250420-lfy-033

试题类型：单选题

标准答案：D

试题难度：容易

试题解析：strftime()方法中直接写入汉字会报错（'locale' codec can't encode character '\u65f6' ...）。%y 表示两位数的年份，%m 表示两位数的月份，%d 表示两位数的日期，%H 表示 24 小时制下的小时数，%M 表示两位数的分钟数，%S 表示两位数的秒数。

考生答案：D

考生得分：2

是否评分：已评分

评价描述：

15. 某 Python 程序段如下：

```
a=[0]*5
for i in range(5):
    num=random.randint(1,10)
    if num%2==0:
        a[i]=num
```

程序运行后，下列输出的列表 a 不可能的是？（ ）

- A. [0, 0, 0, 0, 0]
- B. [2, 4, 5, 6, 11]
- C. [0, 2, 2, 0, 10]
- D. [8, 4, 6, 4, 0]

试题编号：20250420-lfy-034

试题类型：单选题

标准答案：B

试题难度：容易

试题解析：random.randint(1,10)生成[1,10]之间的整数，但只有偶数才能存入列表 a 中。选项[0, 0, 0, 0, 0]正确，可能五次循环都生成了奇数；选项[2, 4, 5, 6, 11]错误，奇数 5 不可能存入 a 中，且 11 超出随机数生成范围；其他两项正确，a 中结果为 0 的当次循环生成了奇数。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

16. 小阳正在用 Python 语言编写一个程序，用于处理一组学生的成绩数据。他使用列表存储成绩，并尝试提取所有偶数位置的成绩（从 0 开始计数）。以下是他的代码：

```
scores = [96, 87, 72, 66, 99]
print(scores[1::2])
```

请问这段代码的输出结果是什么？（ ）

- A. [96, 72, 99]
- B. [87, 66]
- C. [96, 87, 72]
- D. [66, 99]

试题编号：20250430-cqf-01

试题类型：单选题

标准答案：B

试题难度：容易

试题解析：切片操作 scores[1::2]表示从索引 1 开始，每隔一个元素取一个。因此输出为 [87, 66]。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

17. 有两个列表，一个包含学生的名字 names = ["李明", "王华", "张亮"]，另一个包含他们的年龄 ages = [12, 10, 13]。使用 zip() 函数将这两个列表合并为一个包含元组的列表，每个元组包含一个名字和对应的年龄。在下列 4 个选项中，哪个选项中的代码补全在①处后，可以正确的实现此功能？（ ）

```
names = ["李明", "王华", "张亮"]
ages = [12, 10, 13]
merged_list = ①
print(merged_list)
```

- A. list(zip(names, ages))
- B. zip(names, ages)
- C. list(merge(names, ages))
- D. list(zip(ages, names))

试题编号：20250430-cqf-02

试题类型：单选题

标准答案：A

试题难度：较难

试题解析：

选项 A 正确地使用了 `zip()` 函数将 `names` 和 `ages` 合并，并用 `list()` 函数将结果转换为列表。

选项 B 使用了 `zip()` 函数，但没有将其转换为列表，因此返回的是一个 `zip` 对象，不是列表。

选项 C 中的 `merge()` 函数不存在，Python 中用于合并列表的函数是 `zip()`。

选项 D 虽然使用了 `zip()` 函数，但参数顺序错误，这将导致年龄和名字不匹配。

考生答案：C

考生得分：0

是否评分：已评分

评价描述：

18. 假设有一个列表 `sports = ["football", "run", "basketball"]`，需要向这个列表中添加一个新的元素 "pingpang"。下列哪个选项里的代码执行最快？（ ）

`sports = ["football", "run", "basketball"]`

A. `sports = sports + ["pingpang"]`

B. `sports[3] = "pingpang"`

C. `sports.append("pingpang")`

D. `sports.insert(0, "pingpang")`

试题编号：20250430-cqf-04

试题类型：单选题

标准答案：C

试题难度：一般

试题解析：

选项 C 使用 `.append()` 方法向列表末尾添加元素，这是最快的方法，因为它直接在列表末尾添加元素，不需要移动其他元素。

选项 A 使用 `+` 运算符连接列表，这需要创建一个新的列表，并将原列表和新元素复制到新列表中，速度较慢。

选项 B 尝试在索引 3 的位置添加元素，但由于列表只有 3 个元素（索引为 0 到 2），这将引发 `IndexError`。

选项 D 使用 `.insert()` 方法在列表开头插入元素，这需要移动列表中所有现有元素，速度最慢。

考生答案：C

考生得分：2

是否评分：已评分

评价描述：

19. 期中考试阅卷系统导出的成绩是按字典数据保存的，陈老师想提取字典中所有学生的分数，并转换为列表。

下列哪个选项中的代码能帮陈老师实现这个想法呢？（ ）

```
scores = {'敖闰': 90, '哪吒': 85, '申公豹': 88}
```

- A. `score_list = list(scores)`
- B. `score_list = list(scores.values())`
- C. `score_list = list(scores.items())`
- D. `score_list = list(scores.keys())`

试题编号：20250430-cqf-05

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：

选项 B 正确地使用了 `scores.values()` 方法来获取字典中的所有值，并用 `list()` 函数将其转换为列表。

选项 A 错误地直接将字典转换为列表，这样会提取字典的键，而不是值。

选项 C 使用 `scores.items()` 方法提取字典中的键值对，并转换为列表，这将创建一个包含元组的列表，而不是只包含值的列表。

选项 D 使用 `scores.keys()` 方法提取字典中的所有键，并转换为列表，这同样不符合题目要求。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

20. 小华正在编写一个程序，用于统计班级学生的成绩分布。他使用字典存储成绩等级和对应的学生人数。以下是他的代码：

```
grades = {"A": 5, "B": 10, "C": 15, "D": 5}
```

```
grades["F"] = grades.get("F", 0) + 3
```

```
print(grades["F"])
```

请问这段代码的输出结果是？（ ）

- A. 0
- B. 3
- C. 5
- D. 10

试题编号：20250430-cqf-20

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：字典 `grades` 中没有键 "F"，因此 `grades.get("F", 0)` 返回默认值 0。然后通过 `+ 3` 将其值设置为 3，因此输出为 3。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

21. 小张正在编写一个程序，用于管理一个餐厅的菜单。现在需要将所有菜品的价格统一提高 2 元。以下是他的代码，请选择正确的语句填到①处，补全代码。

```
menu = {"Burger": 10, "Pizza": 15, "Salad": 8}
```

```
for item in menu:
```

```
    ①  
print(menu)
```

- A. menu[key] += 2
- B. menu[value] += 2
- C. menu[keys] += 2
- D. menu[item] += 2

试题编号：20250430-cqf-21

试题类型：单选题

标准答案：D

试题难度：容易

试题解析：字典类型通过键来访问对应的值。在用 for 语句遍历时，变量 item 遍历的是字典 menu 的键。

考生答案：D

考生得分：2

是否评分：已评分

评价描述：

22. 在 Python 语言中，tuple1 = ('aa', 'bb', 'cc', '22')，则 tuple1.index('22') 的值是？
()

- A. 1
- B. 2
- C. 3
- D. 4

试题编号：20250501-wp-006

试题类型：单选题

标准答案：C

试题难度：容易

试题解析：元组与列表和字符串一样，索引下标从 0 开始，最后一个元素“22”的索引下标是 3。

考生答案：C

考生得分：2

是否评分：已评分

评价描述：

23. 下面 Python 程序运行结果是？ ()

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

```
s[4][0]=7
```

```
print(len(s))
```

- A. 4
- B. 5

- C. 6
- D. 7

试题编号：20250501-wp-007

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：元组中的元素不可修改，但如果元组的元素为列表时可通过操作列表修改，但元组中元素个数未变，故选 B。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

24. 以下哪个选项能正确地计算集合中元素的个数？（ ）

- A. count()
- B. length()
- C. size()
- D. len()

试题编号：20250501-wp-023

试题类型：单选题

标准答案：D

试题难度：一般

试题解析：使用内置函数 len() 可以计算集合中元素的个数。选项 A、B、C 都不是集合的有效方法。

考生答案：D

考生得分：2

是否评分：已评分

评价描述：

25. 执行以下代码后，集合 s 的内容是？（ ）

```
s = {1, 2, 3}
```

```
s.add(3)
```

```
s.update({4, 5})
```

- A. {1, 2, 3,3,4, 5}
- B. {1, 2, 3, 4, 5}
- C. {1, 2, 4, 5}
- D. {3, 4, 5}

试题编号：20250501-wp-024

试题类型：单选题

标准答案：B

试题难度：一般

试题解析：add(3) 会因元素重复被忽略；update({4,5}) 将合并新元素，最终集合为 {1, 2, 3, 4, 5}。

考生答案：B

考生得分：2

是否评分：已评分

评价描述：

二、判断题(共 10 题，共 20 分)

26. 在 Python 中，字符串 `s = "hello"` 的最后一个字符可以通过 `s[-1]` 访问。（ ）

正确 错误

试题编号：2025-zpf-013

试题类型：判断题

标准答案：正确

试题难度：容易

试题解析：Python 支持负数索引，-1 表示最后一个字符。

考生答案：正确

考生得分：2

是否评分：已评分

评价描述：

27. 在 Python 中，字符串可以通过索引直接修改，例如 `s = "cat"; s[0] = "b"` 会将字符串变为 "bat"。（ ）

正确 错误

试题编号：2025-zpf-015

试题类型：判断题

标准答案：错误

试题难度：一般

试题解析：字符串是不可变类型，无法直接通过索引修改字符。

考生答案：错误

考生得分：2

是否评分：已评分

评价描述：

28. 已知 `numbers=[1,2,3,4,5,6,7,8,9,10,11,12,13,14]`

小明编写了以下代码：

```
result = [x for x in numbers if x % 2 == 0 and x > 10]
```

这段代码能正确生成符合条件的列表 [10, 12, 14]。（ ）

正确 错误

试题编号：2025-zpf-030

试题类型：判断题

标准答案：错误

试题难度：较难

试题解析：代码中条件为 `x % 2 == 0 and x > 10`，导致等于 10 的偶数被排除（正确条件应为 `x % 2 == 0 and x >= 10`），实际输出应为 [12, 14]。

考生答案：正确

考生得分：0

是否评分：已评分

评价描述：

29. 在 python 中，range 函数生成的整数序列中的元素可以通过索引访问。（ ）

如：

```
>>> r=range(1,8,2)
```

```
>>> r[1]
```

```
3
```

正确 错误

试题编号：20250420-lfy-019

试题类型：判断题

标准答案：正确

试题难度：容易

试题解析：range(1,8,2)产生的整数序列是[1,3,5,7]，索引从 0 开始，所以 r[1]的返回值为 3。

考生答案：正确

考生得分：2

是否评分：已评分

评价描述：

30. 在 Python 中用 wordcloud 库绘制词云，其中有句代码为：

```
w=wordcloud.WordCloud(background_color="white",height=450,width=300)
```

。

最终词云生成在高为 450 像素，宽为 300 像素的白色长方形中。（ ）

正确 错误

试题编号：20250420-lfy-035

试题类型：判断题

标准答案：正确

试题难度：一般

试题解析：wordcloud 库的常规方法中，background_color 为指定词云图片的背景颜色，white 为白色；height 为指定词云对象生成的高度，单位像素；width 为指定词云对象生成图片的宽度，单位像素；mask 为指定词云形状，题目中没有说明则默认为长方形。

考生答案：正确

考生得分：2

是否评分：已评分

评价描述：

31. 在 Python 中，因为字典 members 中有重复的值，所以运行下面的代码会报错。

（ ）

```
members = {  
    "001": "Alice",  
    "002": "Bob",  
    "003": "Alice"  
}  
print(members["003"])
```

正确 错误

试题编号：20250430-cqf-22

试题类型：判断题

标准答案：错误

试题难度：一般

试题解析：程序不会报错。在 Python 字典中，键必须是唯一的，而值可以重复。虽然"Alice"作为值重复了两次，但键（会员 ID）是唯一的，因此字典可以正常存储这些数据。members["003"]会返回键"003"对应的值"Alice"。

考生答案：错误

考生得分：2

是否评分：已评分

评价描述：

32. 列表 L=[1,2,3,4,5],则表达式 L[::-1][-1]的结果与 L[0]相同。 ()

正确 错误

试题编号：20250501-wp-005

试题类型：判断题

标准答案：正确

试题难度：一般

试题解析：L[::-1][-1]的结果为 1。

考生答案：错误

考生得分：0

是否评分：已评分

评价描述：

33. 元组 t=([1,2],'a','b')，执行语句 t[0][2]=3，元组会更改为：t= ([1,2,3],'a','b')。()

正确 错误

试题编号：20250501-wp-008

试题类型：判断题

标准答案：错误

试题难度：一般

试题解析：元组中的列表元素可以追加用 t[0].append(3)，直接用下标追加报错。

考生答案：错误

考生得分：2

是否评分：已评分

评价描述：

34. 在 Python 中，集合中的元素可以是列表。()

正确 错误

试题编号：20250501-wp-025

试题类型：判断题

标准答案：错误

试题难度：一般

试题解析：集合中的元素必须是可哈希的（不可变类型），列表是可变类型，不能作为集

合的元素。

考生答案：错误

考生得分：2

是否评分：已评分

评价描述：

35. 集合可以使用数学运算符进行交集、并集等运算。（ ）

正确 错误

试题编号：20250501-wp-026

试题类型：判断题

标准答案：正确

试题难度：一般

试题解析：集合支持使用数学运算符（如&、|、-）进行交集、并集、差集等运算。

考生答案：正确

考生得分：2

是否评分：已评分

评价描述：

三、编程题(共 3 题，共 30 分)

36. 传感器采集数据

在人工智能的多模态数据处理中，不同传感器采集的特征数据需要融合。每个传感器用唯一 ID 标识，数据格式为“ID:特征值”。

编写程序将多个传感器的特征值合并，并计算每个 ID 的加权调和平均数（权重由出现次数决定）。给定若干行传感器数据（格式为 ID:值），请按以下规则处理：

数据去重：同一 ID 的重复数据仅保留最后一次出现的值；

权重计算：统计每个 ID 的数据出现总次数；

融合计算：输出每个 ID 的值*出现次数（即权重与最后值的乘积）。

例如：

输入：

TempS:20

PressS:150

TempS:25

HumS:60

输出：

HumS:60

PressS:150

TempS:50 # TempS 出现 2 次，最后值为 25 → 25*2=50

下面是小明编写的 Python 代码，请你将空白内容补充完整。

sensor_info = {} # 存储每个传感器 ID 的最后值和出现次数

while True:

 data_line = input()# 获取用户输入的一行数据

 if data_line == 'p':# 假设当输入 p 时停止输入

 break

 parts = data_line.split(':')# 把输入的行按冒号分割成 ID 和值

 sensor_id = parts[①]

```

sensor_value = float(parts[1])
if sensor_id in sensor_info: # 检查这个 ID 是否已经在字典里
    sensor_info[sensor_id][0] = sensor_value
    sensor_info[sensor_id][1] = sensor_info[sensor_id][②] + 1 # 统计出现次数
else:
    sensor_info[sensor_id] = [sensor_value, 1]

if len(sensor_info) == 0: # 检查字典里是否没有数据

    print("NULL")

else:

    sorted_ids = sorted(③) # 对字典里的 ID 进行排序
    for current_id in ④:
        last_value = sensor_info[current_id][0]
        count = sensor_info[current_id][1]
        # 计算加权值
        weighted_value = ⑤
        print(f'{current_id}:{weighted_value}')

```

试题编号：202506-P5-36

试题类型：编程题

标准答案：

参考程序：

```

# 初始化一个空字典，用来存储每个传感器 ID 的最后值和出现次数
# 字典的键是传感器 ID，值是一个列表，列表的第一个元素是最后值，第二个元素是出现次数
sensor_info = {}
while True:
    # 获取用户输入的一行数据
    data_line = input()
    # 当输入为 EOF 时，结束输入循环
    # 假设当输入 p 时停止输入
    if data_line == 'p':
        break
    # 把输入的行按冒号分割成 ID 和值
    parts = data_line.split(':')
    sensor_id = parts[0]
    sensor_value = float(parts[1])
    # 检查这个 ID 是否已经在字典里

```

```

if sensor_id in sensor_info:
    # 如果在，更新最后值为当前值
    sensor_info[sensor_id][0] = sensor_value
    # 并且把出现次数加 1
    sensor_info[sensor_id][1] = sensor_info[sensor_id][1] + 1
else:
    # 如果不在，在字典里添加这个 ID，并记录初始值和出现次数为 1
    sensor_info[sensor_id] = [sensor_value, 1]
# 检查字典里是否没有数据
if len(sensor_info) == 0:
    # 如果没有数据，输出 NULL
    print("NULL")
else:
    # 对字典里的 ID 进行排序
    sorted_ids = sorted(sensor_info.keys())
    # 遍历排序后的 ID
    for current_id in sorted_ids:
        # 获取当前 ID 的最后值和出现次数
        last_value = sensor_info[current_id][0]
        count = sensor_info[current_id][1]
        # 计算加权值
        weighted_value = last_value * count
        # 输出结果
        print(f'{current_id}:{weighted_value}')

```

试题难度：较难

试题解析：

评分标准：

- (1) 0（或等效答案）；（2分）
- (2) 1（或等效答案）；（2分）
- (3) sensor_info.keys()（或等效答案）；（2分）
- (4) sorted_ids（或等效答案）；（2分）
- (5) last_value * count（或等效答案）。（2分）

展示地址：点击浏览

考生答案：（此题已作答）

考生得分：10

是否评分：已评分

评价描述：

37. 购物车系统

派森同学用 Python 语言写了一个简易的购物车系统。这个系统可以存储用户添加到购物车中的商品名称和价格，用空格隔开。用户输入结束后，可以按价格从高到低打印输出购物车清单，最后输出一共有几种商品和总价。请补全横线上的代码，实现该功能。

```

shopping_cart = [] # 初始化购物车列表
print("欢迎使用购物车系统！")
print("请输入商品名称和价格，输入'结束'以停止添加。")
while True:
    item = input("请输入商品名称和价格（用一个空格隔开）：")
    if item == '结束':
        break
    name, price = item.split()
    shopping_cart.append((name.strip(), float(price.strip()))) # 将商品名称和
    价格以元组的形式添加到购物车列表里
shopping_cart.sort(key=lambda x: x[1], reverse= ) # 按价格从高到
    低排序
print("\n 购物车内容（按价格从高到低）：")
# 输出购物车内容
for item in shopping_cart:
    print(f"商品名称: { }, 价格: {item[1]:.2f}")
# 计算总价
total_price = sum(item[1] for item in shopping_cart)
# 输出一共有几种商品和总价
print(f"\n 一共有 { } 种商品，总价为 ￥ {total_price:.2f}")

```

—————示例运行结果—————

```

欢迎使用购物车系统！
请输入商品名称和价格，输入'结束'以停止添加。
请输入商品名称和价格（用一个空格隔开）：笔记本 3.5
请输入商品名称和价格（用一个空格隔开）：中性笔 8
请输入商品名称和价格（用一个空格隔开）：结束
购物车内容（按价格从高到低）：
商品名称: 中性笔, 价格: 8.00
商品名称: 笔记本, 价格: 3.50
一共有 2 种商品，总价为 ￥ 11.50

```

试题编号：202506-P5-37

试题类型：编程题

标准答案：

参考程序：

初始化购物车列表

```

shopping_cart = []
# 提示用户输入商品信息
print("欢迎使用购物车系统！")
print("请输入商品名称和价格，输入'结束'以停止添加。")
# 循环收集用户输入的商品信息

```

```

while True:
    item = input("请输入商品名称和价格（用一个空格隔开）：")
    if item == '结束':
        break
    name, price = item.split()
    shopping_cart.append((name.strip(), float(price.strip()))) # 将商品名称和价格
以元组的形式添加到购物车列表里
shopping_cart.sort(key=lambda x: x[1], reverse=True) # 按价格从高到低排序
print("\n 购物车内容（按价格从高到低）：")
# 输出购物车内容
for item in shopping_cart:
    print(f"商品名称: {item[0]}, 价格: {item[1]:.2f}")
# 计算总价
total_price = sum(item[1] for item in shopping_cart)
# 输出一共有几种商品和总价
print(f"\n 一共有 {len(shopping_cart)} 种商品，总价为 ￥ {total_price:.2f}")

```

—————示例运行结果—————

欢迎使用购物车系统！
请输入商品名称和价格，输入'结束'以停止添加。
请输入商品名称和价格（用一个空格隔开）：笔记本 3.5
请输入商品名称和价格（用一个空格隔开）：中性笔 8
请输入商品名称和价格（用一个空格隔开）：结束
购物车内容（按价格从高到低）：
商品名称: 中性笔, 价格: 8.00
商品名称: 笔记本, 价格: 3.50
一共有 2 种商品，总价为 ￥11.50

试题难度：较难

试题解析：

评分标准：

- (1) append（或等效答案）；（2分）
- (2) True 或 1（或等效答案）；（2分）
- (3) item[0]（或等效答案）；（3分）
- (4) len(shopping_cart)（或等效答案）。（3分）

展示地址：点击浏览

考生答案：（此题已作答）

考生得分：7

是否评分：已评分

评价描述：

38. 字符串循环移位

字符串的循环移位过程是将字符串的第一个字符移动到末尾形成新的字符串，例如，现有一个字符串"AABCD"，移位一次后产生新字符串"ABCDA"，再移位一次产生新字符串"BCDAA"，依次类推。下面的程序实现的功能是：首先输入两个字符串，判断其中较短

的字符串是否是较长的字符串通过若干次循环移位后的新字符串的子串。例如输入字符串 "CDAA" 和 "AABCD"，因为 "CDAA" 是由 "AABCD" 两次移位后产生的新字符串 "BCDAA" 的子串，所以结果输出 "Yes"。如果输入字符串 "ABCD" 和 "ACBD"，因为 "ABCD" 不是 "ACBD" 循环移位产生的新字符串的子串，"ACBD" 也不是 "ABCD" 循环移位产生的新字符串的子串，所以输出 "No"。

实现上述功能的 Python 程序如下，请在划线处填入合适的代码。

```
a = input("请输入一个字符串:")
b = input("请输入另一个字符串:")
if len(a) < len(b):
    a, b = b, a
flag = False # 初始化匹配标志为 False
for i in range(len(a)): # 遍历长字符串的所有可能起始位置
    p = i # 初始化当前检查开始位置
    ①
    for j in range(len(b)): # 遍历短字符串的每个字符
        if b[j]==a[p%len(a)]:
            count+=1
            ② # 移动到下一个位置
    if count==len(b): # 如果完全匹配短字符串长度
        print("Yes")
        ③
        break
if ④:
    print("No")
```

试题编号：202506-P5-38

试题类型：编程题

标准答案：

参考程序：

```
a = input("请输入一个字符串:")
b = input("请输入另一个字符串:")
if len(a) < len(b):
    a, b = b, a
flag = False
for i in range(len(a)):
    p = i
    count=0
    for j in range(len(b)):
        if b[j]==a[p%len(a)]:
            count+=1
            p += 1
    if count==len(b):
        print("Yes")
```

```
        flag = True
        break
if flag == False:
    print("No")
```

试题难度：较难

试题解析：

评分标准：

- (1) count=0 (或等效答案) ； (3 分)
- (2) p+=1 (或等效答案) ； (3 分)
- (3) flag==True (或等效答案) ； (2 分)
- (4) flag==False (或等效答案) 。 (2 分)