

考试说明

1. 考试时间：合计 120 分钟，允许提前交卷。
2. 考试过程中，不能连接未经指定网站或服务器。
3. 开卷考试部分，可以查阅纸质文档，不能查阅除 Python 编程环境自带帮助文件以外的任何类型的电子文档。
4. 考试过程中，不得使用任何形式的电子存储设备，不可使用手机。

编程题注意事项与提交方式（80 分，开卷）

5. Sample.py 中的函数名、参数数量和顺序不可以修改。
6. 调用自己写的函数、自己测试的代码等，请写入到 `if __name__=="__main__":`，不要写到全局环境中。
7. 不要在全局环境中调用 `input()`。
8. 每个函数中不需要 `print`，而是用 `return` 返回结果。
9. 不要使用关键字（`if`、`else`、`break`、`def` 等，会导致语法错误）作为自己的变量名，也不要使用内嵌名字（如 `list`、`int`、`input` 等，没有语法错误，但是很危险）作为自己的变量名，例如：

`list = []`
10. 代码中不要出现任何中文。
11. 确保提交的时候，自己的程序可以正常运行，不要遗留任何语法错误。
12. 所有脚本程序内容必须仅包含在一个脚本程序文件（py 文件）中。
13. 提交前务必关闭 IDLE 或 PyCharm 编程环境。
14. 在浏览器的地址栏中输入 `http://192.168.125.3`，点击相应链接进入提交页面。
15. 按要求输入两遍自己的学号。
16. 点击“选择文件”按钮，选择自己的脚本程序文件。点击“提交”按钮提交。
17. 如提交成功，系统会显示相关信息。如果提交不成功，请重复步骤 16-18。
18. 提交成功后，可点击“查看内容”按钮检查提交的内容。

答题方式示意图：

19. 请根据图 1、图 2 和图 3 所示的说明严格规范源文件结构。

```
1  #考试空文件
2
3  import math
4
5  def func2(a,b):
6      return
7
8  def func3(1st):
9      return
10
11  if __name__=="__main__":
12      pass
13
```

图 1 空文件样例

```

1  #考试标准答案文件
2
3  import math
4
5  def func2(a,b):
6      if a<=0 or b<=0:
7          return
8      if a>b:
9          a, b = b, a
10     s = 0
11     for i in range(a, b+1):
12         n = i
13         h = 0
14         while i>0:
15             h = h * 10 + i % 10
16             i = i // 10
17         if h == n:
18             s = s + 1
19     return s
20
21
22 def func3(lst):
23     for i in range(len(lst)-1, -1, -1):
24         if lst[i]<0 or lst[i]%3==0:
25             lst.remove(lst[i])
26     lst.sort(reverse=True)
27     return lst
28
29
30 if __name__=="__main__":
31     print(func2(121,121))
32     print(func2(12221,12221))
33     print(func2(0,10))
34     print(func2(1,10))
35     print(func2(2,100))
36
37     print(func3([7,123,1,2,3,-1,66]))
38     print(func3([]))
39     print(func3([0]))

```

每道题目的代码包含在一个确定名称的函数体内。函数中不包含 input 函数和 print 函数。

测试代码写在此处，且全部包含的 if 语句体内，即保持相同的缩进位置。

图 2 正确的答题文件格式

```
1 #考试错误答案文件
2
3 import math
4
5 def func2(a,b):
6     if a<=0 or b<=0:
7         return
8     if a>b:
9         a, b = b, a
10    s = 0
11    for i in range(a, b+1):
12        n = i
13        h = 0
14        while i>0:
15            h = h * 10 + i % 10
16            i = i // 10
17        if h == n:
18            s = s + 1
19    return s
20
21 print(func2(121,121))
22
23 def func3(lst):
24     for i in range(len(lst)-1, -1, -1):
25         if lst[i]<0 or lst[i]%3==0:
26             lst.remove(lst[i])
27     lst.sort(reverse=True)
28     return lst
29
30 print(func3([7,123,1,2,3,-1,66]))
31
32
33 if __name__=="__main__":
34
35     print(func2(12221,12221))
36     print(func2(0,10))
37     print(func2(1,10))
38     print(func2(2,100))
39     print(func3([]))
40     print(func3([0]))
41
42 print(func2(2,100))
43 print(func3([]))
```

错误!!! 不要在函数之间插入测试代码

错误!!! 不要在函数之间插入测试代码

错误!!! 不要在 if 语言体外插入测试代码

图 3 错误的答题文件格式

题目说明

1. 给定整数 m 和 n ，如果 m 和 n 都大于 1，则判定 m 和 n 是否互质，并返回判定结果。

相关说明	
输入条件	输入参数 m 和 n 是整数。大小关系未知。
输出要求	如果 m 和 n 中任何一个小于或等于 1，则返回 <code>None</code> ，否则判定两数是否互质。 如果 m 和 n 互质，则返回布尔值 <code>True</code> ，否则返回布尔值 <code>False</code> 。
其它要求	将代码写入函数 <code>func1</code>

测试用例：

输入	返回
2,3	<code>True</code>
4,8	<code>False</code>

2. 一个整数列表 $L=[a_1, a_2, \dots, a_n]$ 中，如果一对数 (a_i, a_j) 满足 $a_i > a_j$ 且 $i < j$ ，那么这对数就称为一个逆序，列表 L 中逆序的数量称为逆序数。求一个整数列表 L 的逆序数。

相关说明	
输入条件	列表中的元素都是整数
输出要求	如果 L 为空或者 L 中只有一个元素，返回 0，否则返回 L 的逆序数。
其它要求	将代码写入函数 <code>func2</code>

测试用例：

输入	返回
[4,3,2,1]	6
[1,3,2,4]	1

3. 矩阵相乘： 输入两个整数类型的矩阵 `mat1` (`m` 行 `d` 列) 和 `mat2` (`d` 行 `n` 列)，返回矩阵相乘后的结果 `mat1*mat2` (`m` 行 `n` 列)。矩阵均用二维列表进行表示。

相关说明	
输入条件	两个矩阵分别严格满足 $m*d$ 和 $d*n$ 的形状 ($m \geq 1, d \geq 1, n \geq 1$ ，具体数值需要根据输入确定)，矩阵中的元素均为整数。
输出要求	返回相乘后的矩阵，用二维列表表示，每一个元素均为整数
其它要求	将代码写入函数 <code>func3</code>

测试用例：

输入	返回
[[1,2]] [[1],[2]]	[[5]]
[[1,2],[1,3]] [[1,1],[1,0]]	[[3,1],[4,1]]

4. 一维列表转成二维列表： 输入一个长度为 $n*n$ 的一维列表， 返回一个 n 行 n 列的二维列表。

相关说明	
输入条件	一维列表能保证长度是 $n*n(n \geq 1)$ ，具体数值需要根据输入确定)，且每个元素为整型。
输出要求	转换后的二维列表
其它要求	将代码写入函数 <code>func4</code>

测试用例：

输入	返回
[1]	[[1]]
[2,1,3,4]	[[2,1],[3,4]]

5. 给定一个字符串，包含了若干个以空格分开的单词，统计其中每个单词出现的次数，以列表的形式返回其中出现次数最多的三个单词（三者按照出现次数降序排序，当出现次数相同时，对单词按照字典序降序排序），如果不足三个单词，则按照上述规则排序后全部返回。

相关说明	
输入条件	一个只包括西文字符的字符串。
输出要求	返回一个元素是字符串的列表
其它要求	将代码写入函数 <code>func5</code>

测试用例：

输入	返回
'hello hi hello apple'	['hello', 'hi', 'apple']
'a'	['a']

6. 仅包含小写字母的两个单词 S 和 T 的 Jaccard 系数（记为 J ）由如下三个统计量来确定：令 a 是在两个单词中都出现的字母的个数， b 是在 S 中出现但没有在 T 中出现的字母的个数， c 是在 T 中出现但没有在 S 中出现的字母的个数，那么 $J = a / (a + b + c)$ 。给定两个单词 S 和 T ，求确定其 Jaccard 系数的三个统计量 a, b, c 。

相关说明	
输入条件	两个仅包含小写字母的单词
输出要求	以元组形式返回三个统计量，即 (a, b, c)
其它要求	将代码写入函数 <code>func6</code>

测试用例：

输入	返回
'his', 'she'	(2,1,1)
'hello', 'python'	(2,2,4)

7. 统计一个非空字符串中出现次数最多的字符及其出现次数。其中英文字母不区分大小写，全部统计为大写字母，如'a'和'A'在计数时进行合并为'A'。结果以包含字符和对应次数的列表形式进行返回。**数据中不存在并列最多的情况，该情况不需要考虑。**

相关说明	
输入条件	能保证目标字符串非空、且其中不存在出现次数并列最多的字符
输出要求	结果以包含字符和对应次数的列表形式进行返回。
其它要求	将代码写入函数 func7

测试用例：

输入	返回
'1aA'	['A',2]
'a'	['A',1]

8. 一个字符串中存在多个正整数，请提取出位数在[3,5]之间的所有正整数，构成一个列表，对此列表按照数字和平均值（各位数字的总和/位数）进行降序排序，并返回排序结果列表。数字和平均值就是各位数字的总和除以位数，例如 2345 的数字和平均值 $= (2+3+4+5)/4=3.5$ ，12 的数字和平均值 $= (1+2)/2=1.5$ 。

相关说明	
输入条件	存在多个正整数的字符串
输出要求	结果以满足要求的列表形式进行返回。如原字符串中不存在满足条件的正整数，返回 None
其它要求	将代码写入函数 func8

测试用例：

输入	返回
'123a4567 1'	[4567,123]
'1234'	[1234]