

COMP9021 2016S2考题解析



更多复习资料
及学习视频：



报PTE送期末班，
报期末班送
Assignment讲解

登登教育



登登教育 IT补习 和 PTE 联袂巨献

暑期学PTE送任意一门期末冲刺班

COMP6714、COMP9020、COMP9021、
COMP9024、COMP9311、COMP9313、
COMP9417、COMP9444、COMP9331、

学员优惠福利

新学员报PTE享九折优惠：\$899

老学员报PTE享七折优惠：\$699

79分保过班

三人同行，再享20刀学费减免
五人同行，再享50刀学费减免
马上呼朋唤友，组团报名，转发朋友圈再减10刀

登登教育

期末冲刺复习班



课程	主讲	考前答疑	价格
COMP 6714	8小时	4小时	230刀/人
COMP 9020	8小时	4小时	230刀/人
COMP 9021	8小时	8小时	230刀/人
COMP 9024	8小时	4小时	230刀/人
COMP 9311	8小时	4小时	230刀/人
COMP 9313	8小时	4小时	230刀/人
COMP 9417	8小时	4小时	230刀/人
COMP 9444	8小时	4小时	230刀/人
COMP 9331	8小时	4小时	230刀/人

期末冲刺复习精品课程，原价**230刀/人**

三人同行，团购价只需**220刀/人**

五人同行，团购价只需**200刀/人**

任选三门课程，尊享特价**570刀/人**

马上呼朋唤友，组团报名，转发朋友圈再减**10刀!!!**

独家福利



需要了解：

- 理解题意，判断考察的是哪个知识点
- 在打印输出时，复制给出的标准输出，最好不要自己敲
- 当有特定的case报错时，可以将最后两行代码注释掉，将该case放在 `if __name__ == '__main__':` 下进行测试
- 做题过程中可以导入不同模块，例如 `collections`, `math`, `itertools`
- 如果不需要定义函数或导入模块，在 `# Insert your code here` 下写自己的代码，不要修改给定的代码

Sample 1

- 考察知识点：列表的重新排序
- 导入新模块：无
- 新建一个list，首先判断原来列表是否存在元素，如存在，在新列表中添加第一个元素
- 再从第二个元素开始遍历列表，判断前后大小关系，依次添加

```
result = []
if L:
    result.append([L[0]])
    first = L[0]
    for second in L[1:]:
        if first < second:
            result[-1].append(second)
        else:
            result.append([second])

    first = second

print(f"The decomposition of L into increasing sublists is: {result}")
```

Sample 2

- 考察知识点：去除一个数末尾的0，再计算有多少位
- 导入新模块：无
- 需要先判断n是否为0
- 将int类型转化为str类型，strip去除末尾的0再计算字符串的长度

```
if n == 0:
    print(f"{n} factorial is {n_factorial}")
    print(f"It has {len(str(n_factorial).strip('0'))} digit, the trailing 0's excepted")
else:
    print(f"{n} factorial is {n_factorial}")
    print(f"It has {len(str(n_factorial).strip('0'))} digits, the trailing 0's excepted")
```

Sample 3

- 考察知识点：求素数
- 导入新模块：`from math import sqrt`
- 找两个数之间所有的质数，再计算两两之间的差值，找最大的差值作为gap
- 先写求素数的函数，将区间内所有质数添加在一个列表里，相减取最大值

Sample 3

```
def is_prime(n):  
    # Only used to test odd numbers.  
    if n == 1:  
        return False  
    if n == 2:  
        return True  
    if n % 2 == 0:  
        return False  
  
    return all(n % d for d in range(3, round(sqrt(n)) + 1, 2))  
  
primes = []  
for number in range(a, b + 1):  
    if is_prime(number):  
        primes.append(number)  
  
if len(primes) > 1:  
    first = primes[0]  
    for second in primes[1:]:  
        gap = second - first - 1  
        if max_gap < gap:  
            max_gap = gap  
        first = second  
  
print(f"The maximum gap between successive prime numbers in that interval is {max_gap}")
```

Sample 5

- 考察知识点：按照升序降序打印
- 导入新模块：无
- $a \leq i \leq j \leq b$, i 是升序, j 是降序, i 和 j 的平均数是升序或降序
- 定义函数判断一个数包含的数字是否为严格升序或降序, 遍历区间的每一个数, 调用函数分别判断
- $\text{average}, c = \text{divmod}((i + j), 2)$ 求平均数和余数

Sample 5

- 检查升序

```
def check_increasing(n):  
    m = str(n)  
    if m:  
        first = m[0]  
        for second in m[1:]:  
            if first >= second:  
                return False  
            first = second  
        return True  
    else:  
        return False
```

检查降序

```
def check_decreasing(n):  
    m = str(n)  
    if m:  
        first = m[0]  
        for second in m[1:]:  
            if first <= second:  
                return False  
            first = second  
        return True  
    else:  
        return False
```

- 主函数

```
for i in range(a, b + 1):  
    if check_increasing(i):  
        for j in range(i + 1, b + 1):  
            if check_decreasing(j):  
                average, c = divmod((i + j), 2)  
                if c == 0 and (check_increasing(average)  
                               or check_decreasing(average)):  
                    print(f"{i} and {j} with {average} as average")
```