

# 程序设计入门—Python

## 第七周编程作业

### 1.题目内容：

倒排索引（Inverted index），也常被称为反向索引，是一种索引方法，用来存储某个单词存在于哪些文档之中。是信息检索系统中最常用的数据结构。通过倒排索引，可以根据单词快速获取包含这个单词的文档列表。

本作业主要完成以下四个功能：

(1). 建立索引：首先输入 100 行字符串，用于构建倒排索引，每行字符串由若干不含标点符号的、全部小写字母组成的单词构成，每个单词之间以空格分隔。依次读入每个单词，并组成一个由<单词，每个单词出现的行号集合>构成的字典，其中行号从 1 开始计数。

(2). 打印索引：按照字母表顺序依次输出每个单词及其出现的位置，每个单词出现的位置则按行号升序输出。例如，如果“created”出现在第 3, 20 行，“dead”分别出现在 14, 20, 22 行。则输出结果如下（冒号和逗号后面都有一个空格，行号不重复）：

```
...
created: 3, 20
dead: 14, 20, 22
...
```

(3). 检索：接下来输入查询(Query)字符串，每行包含一个查询，每个查询由若干关键字(Keywords)组成，每个关键字用空格分隔且全部为小写字母单词。要求输出包含全部单词的行号（升序排列），每个查询输出一行。若某一关键字在全部行中从没出现过或没有一行字符串包含全部关键字，则输出“None”。遇到空行表示查询输入结束。如对于上面创建的索引，当查询为“created”时，输出为“3, 20”；当查询为“created dead”时，输出为“20”；当查询为“abcde dead”时，输出为“None”；

(4). 高级检索：当输入的 Query 以“AND: ”开始，则执行“与”检索，即要求输出包含全部关键字的行；如果输入的 Query 以“OR: ”开始，则执行“或”检索，即某行只要出现了一个关键字就满足条件。默认情况（不以“AND: ”或“OR: ”开始），执行“与”检索。

输入格式：

首先输入 100 行字符串，每行字符串由若干不含标点符号的、全部小写字母组成的单词构成，每个单词之间以空格分隔。

若干个查询，每个查询占一行，既可能是普通检索，也可能是高级检索。

输出格式：

首先打印索引，然后将每个查询的结果输出到一行。

## 2.题目内容：

实现逆向最大匹配分词算法，即从右向左扫描，找到最长的词并切分。如句子“研究生命的起源”，逆向最大匹配分词算法的输出结果为“研究 生命 的 起源”。

输入格式：

第一行是以 **utf-8** 格式输入的词表，每个词之间以空格分隔。

接下来是若干行以 **utf-8** 格式输入的中文句子。

输出格式：

以 **utf-8** 格式输出的逆向最大匹配的分词结果，每个词之间使用空格分隔。每个输入对应一行输出。

输入样例：

```
你 我 他 爱 北京 天安门 研究 研究生 命 生命 的 起源
研究生命的起源
我爱北京天安门
```

输出样例：

```
研究 生命 的 起源
我 爱 北京 天安门
```

## 第七周编程小测验

**1.**一个学生的信息包括：学号、姓名、年龄、性别、电话。如果想保存一系列学生，并且希望能够快速的查找某一姓名的学生，则下列数据结构哪个更合适？

- A.** 字典                      **B.** 集合                      **C.** 列表                      **D.** 元组

**2.**写出下面程序的输出结果：

```
d1 = {}
d1[2] = 10
d1['2'] = 20
|
d2 = {}
d2[2] = d1
d2['2'] = d2
|
print d2['2']['2']['2']['2'][2][2]
```

---

3. 下列程序的执行结果是：

```
def f2 (my_dict):  
    temp = "  
    for key in my_dict:  
        if temp < key:  
            temp = key  
    return temp
```

```
a_dict = {'bill':1,'rich':2,'fred':10,'walter':20}  
print f2(a_dict)
```

---

4. 一个学生的信息包括：学号、姓名、年龄、性别、电话。如果想保存一系列学生，并且希望能够根据姓名对他们进行排序，则下列数据结构哪个更合适？

- A. 元组                      B. 字典                      C. 字典                      D. 列表

5. 下列程序的输出结果是：

```
def f1 (my_dict):  
    temp = 0  
    for value in my_dict.values():  
        temp = temp + value  
    return temp
```

```
a_dict={'bill':1,'rich':2,'fred':10,'walter':20}  
print f1(a_dict)
```

---

### 第七周编程小测验-答案

1. A                      2. 10                      3. walter                      4. D                      5. 33