词频统计

杨琦
西安交通大学计算机教学实验中心

【例】词频统计

- 输入一系列英文单词(单词之间用空格隔开)，用“xyz”表示输入结束。
- 统计各单词出现的次数（单词不区分大小写），对单词按字典顺序进行排序后输出单词和词频。
- 【运行结果】

请输入一系列英语单词，以xyz表示输入结束
Do you see the star , the little star ? xyz
词频统计结果如下：
,　　1
?　　1
Do　　1

【问题分析】

①数据结构。本题中每个单词有两条信息要记录，一是单词本身，二是单词的出现次数，即使1次，所以可以用结构体。

②查找。每输入一个单词，要在已有单词序列中查找，找到在次数加1，找不到则添加一个新单词，次数置1。

③排序——选择排序

先将待排序序列分成有序部分和无序部分，重复地从无序部分中找出最大的元素，放在有序部分的最后，直到无序部分只有一个元素。如果有N个元素要排序，这样的选择过程只需要N-1次。

【算法描述】查找算法

①输入单词word；

②如果word否为结束标志xyz；转④，否则继续；

③顺序查找word是否在词典中。

- 若已存在词典中，则将对应的词频加1，返回①；
- 若词典中不存在该单词，则向词典中添加新的单词，返回①；

④对词典进行排序；

⑤输出词典内容。

【算法描述】选择排序算法

①设待排序元素用数组A[i]表示，i=0,1,...,N-1；

//控制N-1次选择，每次选择的“最小”元素与A[i]互换

②对i=0，...,N-2

③　　k=i　　//设A[i]是当前最小的元素，它的下标保存在k中

④　　对j=i+1,...,N-1　　//与后面的所有元素比较

　　若A[j]<A[k]，则　　//后面的更小

　　k=j　　//记写最小元素的下标

⑤　　如果k!=i　　//A[i]不是最小的元素

tmp=A[i] , A[i]=A[k], A[k]=tmp　　//交换最小元素和A[i]

⑥N-1次选择后结束，数组A中的元素有序。

【源程序1】

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>          //包含基本输入输出库头文件
#include <cstring>
using namespace std;         //使用名字空间
struct WordList  {           //字典结构体
        char word[20];       //单词
        int freq;            //使用次数
};
int main()  {                //主函数
        WordList list[1000];          //结构体数组
        int N=0;             //实际单词数
```

【源程序2】

```
int i,j,k;                          //循环变量，；临时变量
char tmp[20];                       //临时存放新输入的单词
//------------输入单词-----------------
cout<<"请输入一系列英语单词，以xyz表示输入结束"<<endl;
cin>>tmp;
while(strcmp(tmp,“xyz”)!=0)          {  //不是单词的结束符时循环
        for(i=0;i<N;i++){                //在当前词典中逐个查
                if(strcmp(list[i].word,tmp)==0){
                        list[i].freq++;       //词频加1
                        break;                //不再查找
                }
        }
```

【源程序3】

```
            if(i>=N){    //这时是没有找到，添加该词
               strcpy(list[i].word,tmp);  //添加单词
               list[i].freq=1;            //词频置1
               N++;                       //单词数加1
            }
            cin>>tmp;                     //继续输入单词
}                        //结束时，N 为词典中的单词数
//--------------对词典进行排序--------------
for(i=0;i<N-1;i++){ //控制N-1次选择
   k=i;                          //先设i是当前最小元素的下标，
   for(j=i+1;j<N;j++){           //与后面的单词比较
```

【源程序4】

```
        if(strcmp(list[j].word, list[k].word)<0){
                k=j;                    //记下最小元素的下标
        }
    }
    if(k!=i){                           //最小的下标不是i
        WordList tmp;
        //交换下标是k和i的两个元素
        tmp=list[i];
        list[i]=list[k];
        list[k]=tmp;
    }
```

【源程序5】

```
    }
    //----------输出结果-------------
    cout<<"词频统计结果如下："<<endl;
    for(i=0;i<N;i++)                    //输出
        cout<<list[i].word<<"\t"<<list[i].freq<<endl;
    return 0;
}
```

【运行结果】

请输入一系列英语单词，以xyz表示输入结束

Do you see the star , the little star ? xyz

词频统计结果如下：

```
,      1
?      1
Do     1
little 1
see    1
star   2
the    2
you    1
```

【问题扩展】

本程序主要分为三大块，输入、排序、输出；

不要把所有工作放在一起做，要一步一步来，这样比较清晰。

如果在统计单词过程中，要去掉标点符号的统计，该如何修改？

如果标点符号与单词连在一起，统计结果会有哪些变化，应该如何解决？