



文本文件与输入-例题

统计文本文件中的单词

1. 题目内容与要求

- ▶ 统计一个文本文件中的各个单词数量。
 - ① 假定已经存在一个文本文件“words.txt”，其内容为多个单词，各单词之间以空格分割；
 - ② 打开这个文件，搜索全部单词；
 - ③ 获取各个单词、个数以及单词总数量并显示出来。

2. 算法分析

- ①定义存放单词和个数的单词类；
- ②定义单词类数组；
- ③打开文件读；
- ④循环读取一个单词，如果文件结束转⑥；
- ⑤保存单词及个数，并统计单词总数量；
- ⑥关闭文件对象；
- ⑦显示各单词和个数以及单词总数量。

3. 单词类

```
class wordtype{  
    char word[20];  
    int count;  
};
```

搜索单词模块程序代码(1)

```
int getwords(wordtype *words){  
    ifstream in("words.txt"); // 打开英文文件  
    if( !in){  
        cout<<"文件打开错误！"<<endl;  
        return 1;  
    }  
  
    int n=0;  
    char word[20];  
    int m;
```

搜索单词模块程序代码(2)

```
while(in){  
    in>>word; // 读单词  
    if(!in){ // 文件结尾时退出循环  
        break;  
    }  
    bool flag=false;  
    for(m=0;m<n;m++){  
        if(!strcmp(word,words[m].word)){ // 已存在该单词  
            words[m].count++;           // 原有单词计数加1  
            flag=true;  
            break;  
        }  
    }  
}
```

搜索单词模块程序代码(3)

```
    if(!flag){ // 发现新单词
        words[m].count=1;           // 新单词计数为1
        strcpy(words[m].word,word); // 保存新单词
        n++;                         // 总单词计数加1
    }
}
in.close(); // 关闭文件

return n; // 返回单词个数
}
```


主函数框架程序代码

```
#include <iostream>
#include <fstream>
using namespace std;
int getwords(wordtype *words);           // 单词搜索模块函数声明
int main()
{
    <核心代码在此>
    return 1;
}
<单词搜索模块函数定义在此!>
```

主函数核心程序代码

```
wordtype words[100]={"",0}; // 单词结构体变量定义与初始化
```

```
int n=getwords(words); // 调用获取单词的函数
```

```
cout<<"英文单词统计结果如下："<<endl;
```

```
for(int m=0;m<n;m++)
```

```
{
```

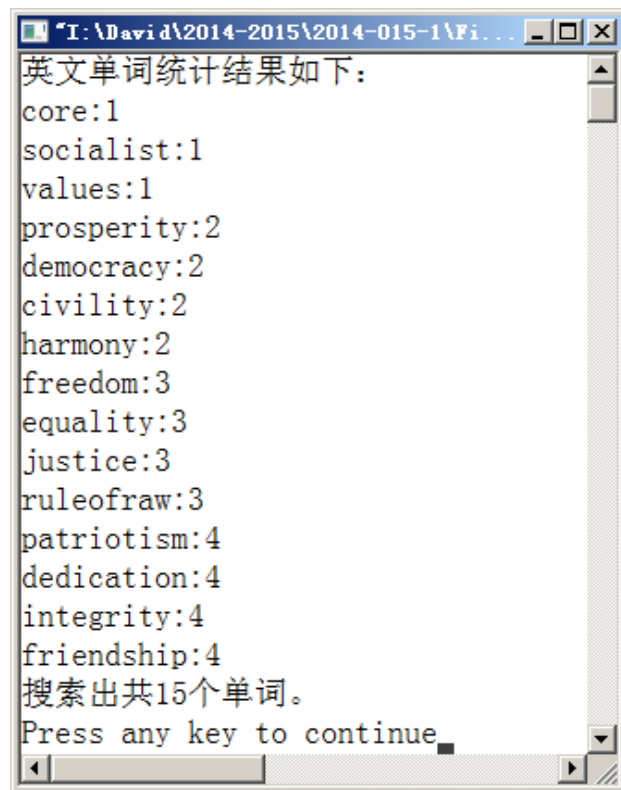
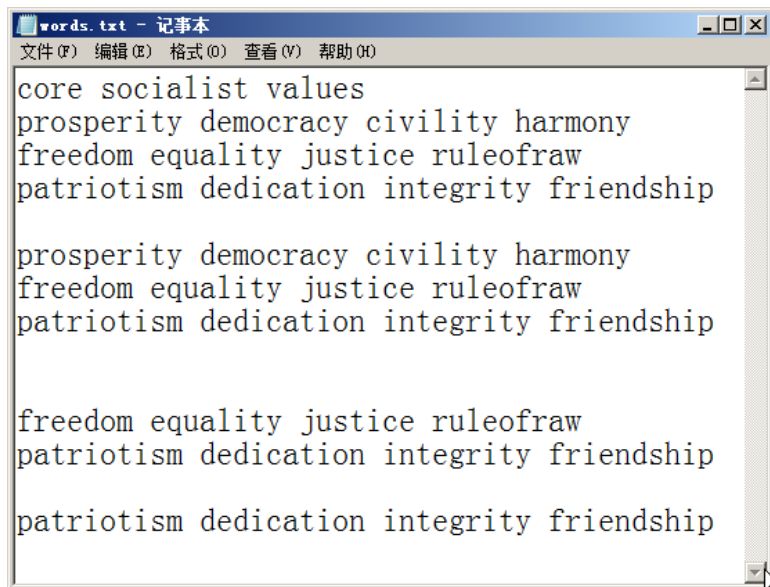
```
    cout<<words[m].word<<':'<<words[m].count<<endl;
```

```
}
```

```
cout<<"搜索出共"<<n<<"个单词。"<<endl;
```



4. 运行结果



5. 延伸思考

问题1



如何对得到的单词数组进行排序？

问题2



单词之间的分割符为其它符号时怎么办？

6. 文本文件输入步骤小结

- ✓ 包含文件流类头文件: `#include<fstream>`
- ✓ 打开文件读: `ifstream in("words.txt");`
- ✓ 读文件: `in>>str;`
- ✓ 用完关闭文件流: `in.close();`

例题的讲解就到这里！

谢谢！

