应用：文件中单词词频统计

应用：文件中单词词频统计

[例] 给定一个英文文本文件，统计文件中所有单词出现的频率，并输出词频最大的前10%的单词及其词频。

假设单词字符定义为大小写字母、数字和下划线，其它字符均认为是单词分隔符，不予考虑。

【分析】关键：对新读入的单词在已有单词表中查找，如果已经存在，则将该单词的词频加1，如果不存在，则插入该单词并记词频为1。

如何设计该单词表的数据结构才可以进行快速地查找和插入？

```
int main() {
    int  TableSize = 10000；  /* 散列表的估计大小 */
    int wordcount = 0,  length;
    HashTable H;
    ElementType word;
    FILE *fp;
    char document[30]= "HarryPotter.txt“; /* 要被统计词频的文件名 */
    H = InitializeTable( TableSize ); /* 建立散列表 */
    if(( fp = fopen(document, “r” ))==NULL)  FatalError(“无法打开文件!\n” );
    while( !feof( fp ) ){
          length = GetAWord( fp, word ); /* 从文件中读取一个单词 */
          if(length > 3){      /* 只考虑适当长度的单词 */
                wordcount++;  /*统计文件中单词总
                InsertAndCount( word, H );
           }
    }
    fclose( fp );
    printf("该文档共出现 %d 个有效单词，", wordcount);
    Show( H ，  10.0/100 );  /* 显示词频前10%的所有单词 */
    DestroyTable( H );       /* 销毁散列表 */
    return 0;
}
```

(1)统计最大词频；
(2)用一组数统计从1到最大词频的单词数；
(3)计算前10%的词频应该是多少
(4)输出前10%词频的单词