**《数据结构1》 实验三 文学研究助手 小组综合性实验**

### 1 实验题目及要求——文学研究助手

#### 1.1问题描述

#### 文学研究人员需要统计英文小说中某些词出现的次数和位置。试编写一个实现这一目标的文字统计系统，称为“文学研究助手”。

#### 1.2基本要求

英文小说存于一个文本文件中，并假设小说中的单词一律不跨行，每行的长度不超过120个字符，待统计的词汇集合要一次输入完毕。要求对英文小说扫描一遍就完成统计工作。

程序的输出结果是每个单词的出现次数和出现位置所在行的行号。其格式自行设计。

#### 1.3输入数据

输入数据包括两部分，第一部分是要统计的单词，不超过 100 个，单词之间用空格分隔；第二部分是被统计的文章，可以考虑把这两部分内容放在一个文件中。例如：以某一C++源程序模拟英文小说，用C++程序设计语言的保留字集作为待统计的词汇集。

## **1.4 输出数据**

对出现在文章中的要统计的单词，输出其在文章中出现的次数和所在的行号。

### 2设计思路及代码

#### 2.1设计思路

由于本题是查找对应文本中的单词的频数以及对应的行数，所以我们可以定义一个哈希表wordMap，键是单词，值是一个pair，其中pair的第一个元素是频数，第二个元素是一个向量，存储该单词出现的行号。在查找的过程中先逐行读取，以空格作为标志分割字符，并将每个单词进行存储。接着处理这些单词，若不为空，则在哈希表wordMap中更新该单词的频数，如果该单词已经存在于哈希表中，则增加其频数；否则，添加新条目，并将当前行号添加到该单词对应的行数向量中。如果要查询对应的单词，只需先从文件中读入字符串，以空格进行分割，再检查哈希表中是否包含这些单词，若包含，则将对应频数与行数输出即可。

#### 2.2算法实现分析与解读

实现该实验的大致算法是：定义一个哈希表wordMap，键是单词，值是一个pair，其中pair的第一个元素是频数，第二个元素是一个向量，存储该单词出现的行号。初始化行号lineNumber为1。使用getline函数逐行读取文件内容到line中。将每一行内容转换为字符串流对象ss，以便按空格进行分割。使用循环按空格分割单词，并将每个单词存储到word中。创建一个新的字符串cleanedWord来存储处理后的单词。这里只保留字母字符，并将其转换为小写。如果处理后的单词不为空（即不是空白字符或标点符号），则在哈希表wordMap中进行如下操作：更新该单词的频数，如果该单词已经存在于哈希表中，则增加其频数；否则，添加新条目。将当前行号添加到该单词对应的行数向量中。在处理完一行后，增加行号lineNumber的值，以便在下一行处理时使用。将从文件中读取的要查找的字符串queryWords转换为字符串流对象querySS，以便按空格分割，使用循环从字符串流中读取查询单词，直到字符串流结束或读取到空格。对于每个查询单词，检查哈希表wordMap中是否包含该单词，如果哈希表wordMap中包含查询单词，则获取该单词在哈希表中的值（频数和行数向量）。输出查询单词，输出频数，遍历行数向量，并输出每行对应的行号。如果哈希表wordMap中不包含查询单词，则输出“Word not found”。

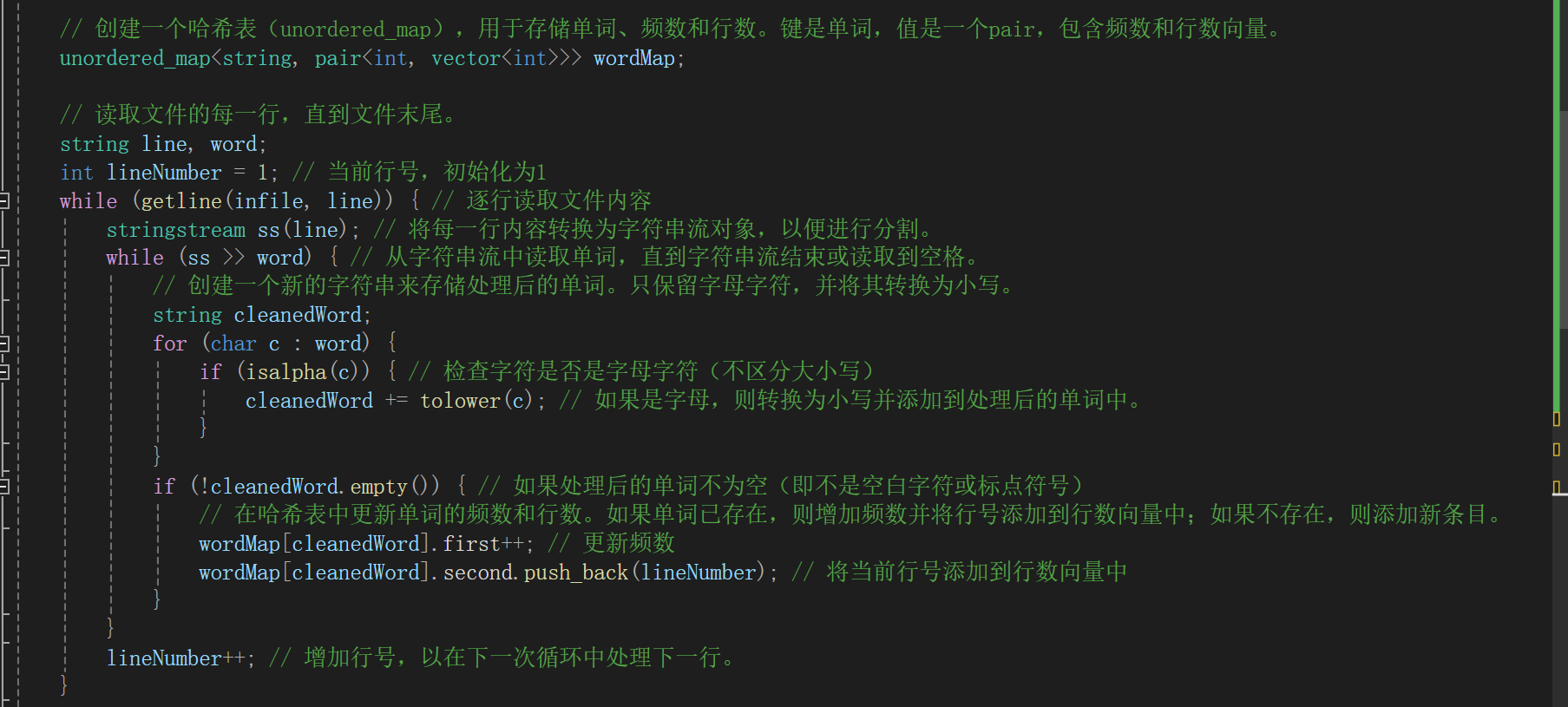


图1 哈希表实现相关功能的代码

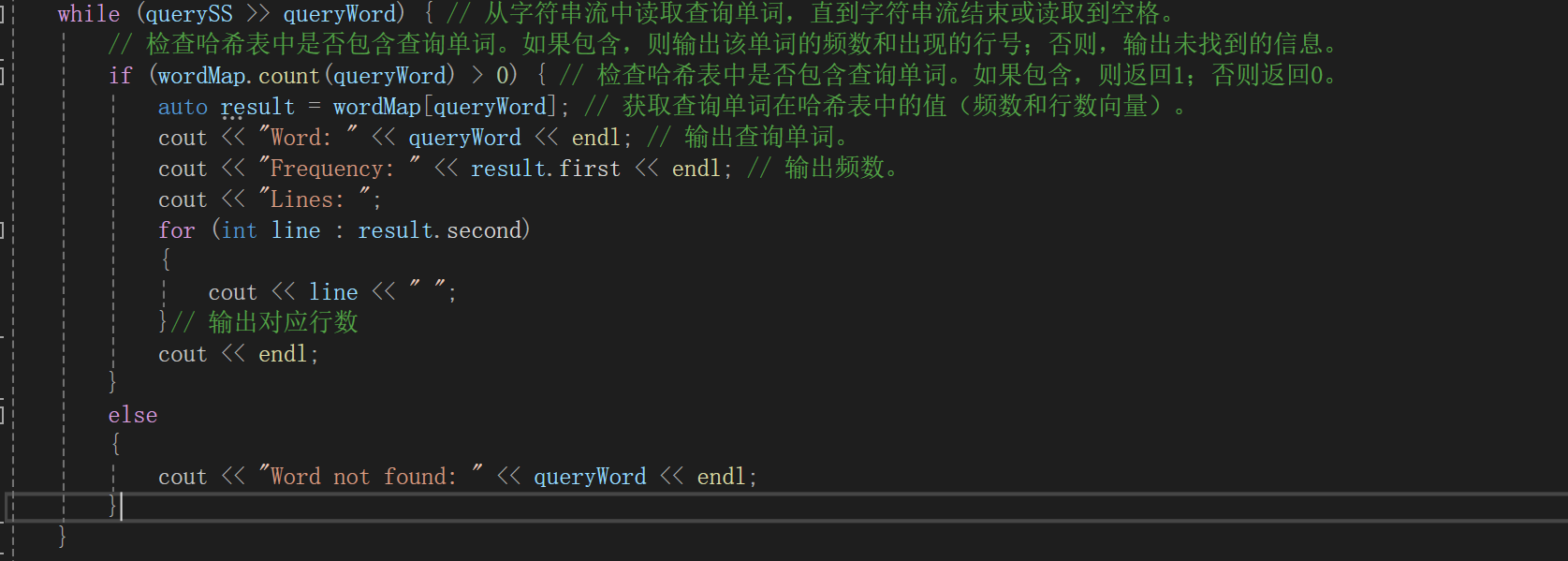


图2 查询具体单词实现代码

#### 2.3测试案例

测试函数具体代码如图3所示。

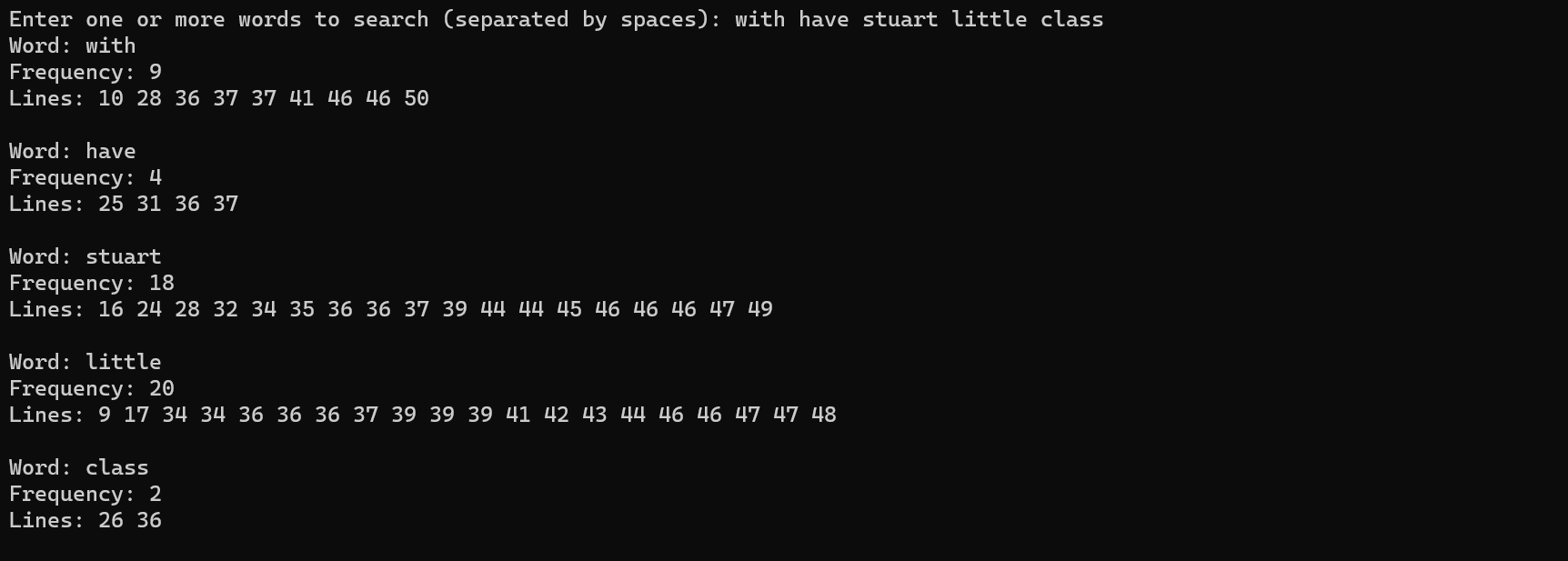


图3 测试结果

根据上述测试案例，所运行的结果正确，顺利完成该实验。

#### 2.4待优化的问题

当处理到“’s”或者其他带单引号的字符串时，会无法识别进而报错；当遇到“-”字符时程序也会无法识别进而报错。

### 3总结和心得

本次实验让我们小组以所学的知识解决了文学研究助手的问题，在此过程中经过不断的讨论研究与实践，加深了对于字符串与哈希表的理解，提升了自己手写代码解决问题的能力，我们也加深理解了字符串的相关处理方法，了解到它的使用场景，也进而掌握了哈希表的相关处理方法，我们了解到了遍历字符串的常用方法，使用for-each循环能够方便地迭代字符串中的每个字符。此外，我们还学会了如何使用条件语句来检查字符是否满足特定条件，例如字母或特殊字符，在处理字符串时，我们还意识到了编码规范的重要性，保持代码整洁和一致的格式可以使代码更易于阅读和维护，另外，在处理字符串时还需要注意字符编码的问题，以避免出现乱码或错误。通过编写这个代码，我们也发现了一些不足之处，比如在判断特殊字符时可能考虑得不够全面，可能需要进一步学习和掌握更多的字符串处理技巧。

虽然在思考的过程中遇到了很多困难，例如读入文件时文件应该保存为ANSI格式、哈希表的预存储空间要充分等，但是经过不断的一起讨论钻研以及对于文献的查阅，以及最后对于代码的不断优化调整，最终解决了问题，我们小组分工明确，在实验上进行了充分的讨论和交流，提出了自己的想法为今后的数据结构的学习做下了铺垫。