上机报告-3

数算B 谢胡睿 2400014151

题目

1.题目背景

在图书馆中，一大重要的任务是如何组织馆藏的文献以及如何提供便捷的查询方式。馆方提供的文献列表是高度组织的格式为“文献编号-文献名”的列表，但是用户在进行搜索时可能会有各种不规范的输入。我们认为文献名是由一系列以空格分隔的词组成的，且在查询中我们只关注关键词。对于任意的查询输入，只要某文献的任何一个关键词为查询的子串，就认为该文献是一个匹配。 **本题综合考查KMP无回溯模式匹配算法与关键词索引表相关知识**

2.题目描述

对于给定的文献列表和查询，输出符合要求的文献编号。

3.输入格式

第一行有一个正整数n，表示总共有 n 条文献信息。接下来的 n 行，每行包括以空格分隔的两个部分：三位文献编号和文献名（文献名可能自然含有空格）。文献编号可能有先导零。接下来的一行，为一个字符串，表示查询的内容，需要从所有输入的关键词中查找查询内容中含有的单词（非关键词表在文末给出）。

4.输出格式

程序运行结束时，将关键词出现在查询内容中的文献的编号输出。第一行为一个正整数n，表示共有n个匹配。接下来n行，每行包括一个三位的文献编号。输出文献编号应当以字典序升序排列，应当包含可能存在的先导零。

5.样例

样例2：

```
#input:
15
001 Introduction to Algorithms
003 Data Structures and Algorithms
007 Advanced Data Structures
012 Computer Networks
015 Operating Systems
020 Database Systems
025 Artificial Intelligence
030 Machine Learning
035 Computer Architecture
040 Software Engineering
045 Cybersecurity
050 Web Development
055 Mobile Application Development
```

```
060 Cloud Computing
065 Big Data Analytics
structuresystemsoftwarepeat

#output:
5
003
007
015
020
040
```

Solution

总体描述

考虑到系统需要处理的数据特点，我选择采用以下数据结构：

文献结构：创建Book类存储文献编号和关键词列表 **停用词表**：使用unordered_set存储所有停用词，便于O(1)时间复杂度的查找 **关键词匹配**：实现KMP算法进行高效的字符串匹配，避免暴力匹配带来的性能问题 **查询结果**：使用vector存储匹配结果，方便排序和输出

设计与实现

见代码

结果

输出

遇到的问题

KMP算法实现错误

问题：初始实现KMP搜索函数时，没有在字符匹配时递增j变量，导致算法无法正确推进，永远无法找到完整匹配。

```
//缺少了这部分
//if (text[i]==word[j]) {
//    j++;
//}
```

解决方案：补充完整KMP算法中字符匹配时的处理逻辑，确保j正确递增

大小写处理

问题：题目要求大小写不敏感，但初始实现没有完全转换输入文本，导致匹配失败。 解决方案：对所有输入文本进行大小写统一处理，将大写字母转换为小写：

总结：

在处理文本分析问题时，停用词过滤、大小写标准化、字符串匹配等环节都需要特别注意。未来在类似项目中，我会更加重视算法正确性的测试和边界情况的处理，确保程序在各种输入下都能正确工作。