



链滴

PAT 甲级刷题实录——1022

作者: [aopstudio](#)

原文链接: <https://ld246.com/article/1581064650839>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原题链接

<https://pintia.cn/problem-sets/994805342720868352/problems/994805480801550336>

思路

题目给出了一些书籍的信息，每本书籍分别包含 ID、标题、作者等信息。输入的是任意种类的书籍信息，要求输出和该信息对应的所有书籍 ID，且要顺序输出。题目中对于输入的信息分别给出了从 1 到 5 的编号代表信息种类，但实际上我们完全不需要知道给的输入到底是什么信息，一个暴力的做法就是建一个 `map<string, set<string>>`，其中第一个 `string` 代表书籍信息字符串，`set<string>` 则存了该书籍信息对应的所有书籍 ID。`set` 也可以改成 `vector`，不过 `vector` 需要手动排序，`set` 是自动排序的。我们在读取书籍信息时，每读取一种信息就将该信息对应的 ID 存入 `set`。其中需要注意的是，键字信息需要一个单词一个单词读取，而其他信息需要整行读取。整行读取的方法是 `getline(cin,s)`，代表字符串变量。`getline` 是默认读到换行符就终止读取的，因此如果读到的第一个字符就是换行符 `getline` 就会读到空字符串。网上有人说 `getline` 不会读取换行符，因此在 `getline` 后面还要加一个 `getchar` 才能继续，我估计他是和其他的哪个方法弄混了。不过我倒发现了 `cin` 方法是不会读取换行符，此在使用 `cin` 方法后如果想使用 `getline` 方法的话，需要在 `cin` 后面加上 `getchar`。另外关于如何读一行中所有的单词而不读到下一行，即读取 `key word` 的过程，网上的大神也给出了方法，代码如下：

```
while (cin >> keyword)
{
    //这里写操作语句
    if (cin.get() == '\n')
        break;
}
```

完整代码

```
#include <iostream>
#include <set>
#include <map>
#include <string>
using namespace std;

int main()
{
    map<string, set<string>> search;
    int N, M;
    cin >> N;
    for (int i = 0; i < N; i++)
    {
        string ID, title, author, keyword, publisher, year;
        cin >> ID;
        getchar();
        getline(cin, title);
        search[title].insert(ID);
        getline(cin, author);
        search[author].insert(ID);
        while (cin >> keyword)
        {
            search[keyword].insert(ID);
            if (cin.get() == '\n')
```

```

        break;
    }
    getline(cin, publisher);
    search[publisher].insert(ID);
    cin >> year;
    search[year].insert(ID);
}
cin >> M;
for (int i = 0; i < M; i++)
{
    string num, word;
    cin >> num;
    getchar();
    getline(cin, word);
    cout << num << ' ' << word<<endl;
    if (search.find(word) != search.end())
    {
        for (set<string>::iterator it = search[word].begin(); it != search[word].end(); it++)
        {
            cout << *it << endl;
        }
    }
    else
    {
        cout << "Not Found" << endl;
    }
}
return 0;
}

```