

L1426 学生统计 ☆☆

题目描述

快开学了，小徐同学想帮老师统计一下本学期学校的学生信息。
老师给了一张学生信息表，每一条数据包括学生姓名、年级、班级。
老师想知道本学期每个年级有多少个班以及多少个人。

输入输出格式

输入：第一行，一个整数 N ，表示有 N 条数据；接下来 N 行，每行包含三个字符串，分别表示姓名、年级、班级。

输出：按年级从小到大，输出年级的班级数和总人数。（注意，年级、班级总数、年级总数之间有一个空格，每行末尾不要有空格）

输入示例	输出示例
5 Tom G7 A Jerry G7 B Llice G7 A Bob G8 A Apple G8 A	G7 2 3 G8 1 2

样例解释

年级	班级列表	班级数	学生人数
G7	A班：Tom、Llice B班：Jerry	2	3
G8	A班：Bob、Apple	1	2

算法分析

- 1. 问题分析
需要统计每个年级的班级数和学生总数。同一个年级的不同班级需要去重统计，而同一个年级的学生人数直接累加。
- 2. 算法选择
 - A. 解法一 - 排序 + 遍历统计：先将所有学生信息按年级和 班级排序，将年级小的放在前面，年级相同的按照班级排序，然后遍历统计每个年级的信息。
 - B. 解法二 - Map/Set 统计：使用 map 和 set 容器自动维护有序性和去重特性，将年级作为键，班级,学生人数作为值
- 3. 实现思路

- 法一：结构体排序 + 遍历统计

- A. 定义结构体存储学生信息（姓名、年级、班级）

```
struct Node {
    string g; 年级
    string c; 班级数
    string n; 总人数
};

bool cmp(Node a, Node b) {
    if (a.g == b.g) return a.c < b.c;
    return a.g < b.g;
}
```

- B. 按**年级升序**、**班级升序**排序

- C. 遍历排序后的数据，统计每个年级的班级数和学生数

```
if (g != a[i].g) { 枚举到了新的年级
    cout << g << " " << cnt_class << " "
        << cnt_stu << endl;
    g = a[i].g, c = a[i].c;
    cnt_class = 1, cnt_stu = 0; 数据清零
}
cnt_stu++; 总人数加1
if (c != a[i].c) {遇到新的班级，班级数加1
    c = a[i].c;
    cnt_class++;
}
```

- 法二：map + set 统计

- A. 使用 map<string, int> 统计每个年级的学生总数

```
map<string, int> s; // 年级 -> 学生数
s[grade]++;
```

- B. 使用 map<string, set<string>> 统计每个年级的班级集合（自动去重）

```
map<string, set<string>> c; // 年级 -> 班级集合
c[grade].insert(clazz);
```

- C. 遍历 map 输出结果

```
for (auto i : s) {
    cout << i.first << " " << c[i.first].size() << " "
        << i.second << endl;
}
```

拓展思考

- 如果要求按**班级数从大到小**输出，该如何修改？
- 如果数据中存在重复记录（相同姓名、年级、班级），该如何去重？