



## 成绩排序

### 【问题描述】

每当考试结束，老师总要对学生的成绩进行排序，以便研究学生学习情况。

老师发现学生数目庞大，所以找来了会编程的你，并且它给你了全部同学的分数，希望你能按如下规则排序：1.分数高的排在前面；2.如果分数相同，就把名字字典序低的放在前面。

### 【输入格式】

第一行一个整数  $n$ 。

接下来  $n$  行每行一个学生名字和一个分数。

### 【输出格式】

每行一个名字和一个分数。

### 【输入样例】

```
4
Xiaozhi 396
Xiaohui 405
Xiaocong 399
Xiaoming 396
```

### 【输出结果】

```
Xiaohui 405
Xiaocong 399
Xiaoming 396
Xiaozhi 396
```

### 【算法分析】

在很多情况下，不是仅仅对一个特征进行排序，而是多个特征排序。

当采用 `sort()` 函数的默认规则排序结构体时，`sort()` 默认结构体中的第一个成员为第一关键字，第二个成员为第二关键字，……，第  $n$  个元素为第  $n$  关键字，然后从小到大排序。

如果要将学生的成绩从大到小排序，当成绩相同时，根据姓名字典序小的优先的规则进行排序，就无法采用默认规则进行排序。

**【参考程序】**

```
#include<iostream>
#include<algorithm>
#include<string>
using namespace std;
struct Student
{
    int score;
    string name;
} student[100];
bool cmp(Student a, Student b)
{
    if (a.score > b.score)
    {
        return true;
    }
    if (a.score < b.score)
    {
        return false;
    }
    if (a.name < b.name)
    {
        return true;
    }
    return false;
}
int main()
{
    int n;
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cin >> student[i].name;
        cin >> student[i].score;
    }
    sort(student, student + n, cmp);
    for (int i = 0; i < n; i++)
    {
        cout << student[i].name << ' ' << student[i].score << endl;
    }
    return 0;
}
```