



## 结构体变量的排序

在 C++ 语言里，已经提供了排序效率很高的内置排序库函数 `sort()`。

使用这个函数，需要

```
1. #include<algorithm>
```

这个函数用来对结构体数组进行排序的调用方式如下：

```
1. //将数组元素 a[m]..a[n-1]按照 cmp 函数定义的比较规则，进行排序
2. // cmp 是对两个元素进行比较，返回值为 bool 类型的函数。返回 true 表示第一个元素在第二
   个元素之前，否则返回 false
3. sort(a+m, a+n, cmp);
```

### 【问题描述】

输入 N 个学生的姓名和语文、数学的得分，按总分从高到低输出，分数相同的按输入先后输出。

### 【输入格式】

第 1 行，有一个整数 N，N 的范围是[1…100]；下面有 N 行，每行一个姓名，2 个整数。姓名由不超过 10 个的小写字母组成，整数范围是[0…100]。

### 【输出格式】

总分排序后的名单，共 N 行，每行格式：姓名 语文 数学 总分。

### 【输入样例】

```
4
yuhong 78 96
jianming 70 99
wei 90 87
lily 78 91
```

### 【输出样例】

```
wei 90 87 177
yuhong 78 96 174
jianming 70 99 169
lily 78 91 169
```

### 【参考程序 1】



```
1. #include<iostream>
2. #include<algorithm>
3. using namespace std;
4. struct Student
5. {
6.     char name[11];
7.     int chinese;
8.     int math;
9.     int total;
10. };
11. Student a[100];
12. //比较函数, s1 比 s2 总分高, 就不用交互顺序
13. bool cmp(Student s1, Student s2)
14. {
15.     return s1.total > s2.total;
16. }
17. int main()
18. {
19.     int n;
20.     cin >> n;
21.     for (int i=0; i<n; i++)
22.     {
23.         cin >> a[i].name >> a[i].chinese >> a[i].math;
24.         a[i].total = a[i].chinese + a[i].math;
25.     }
26.     sort(a, a+n, cmp);
27.     for (int i=0; i<n; i++)
28.     {
29.         cout << a[i].name << ' ' << a[i].chinese << ' ' << a[i].math << ' ' <<
a[i].total << endl;
30.     }
31.     return 0;
32. }
```

### 【参考程序 2】

下面的程序是自己编写的排序过程。

```
1. #include<iostream>
2. #include<algorithm>
3. using namespace std;
4. //定义一个 struct 的类型, 类型名叫: Student
5. struct Student
6. {
```



```
7.     char name[11];
8.     int chinese;
9.     int math;
10.    int total;
11. };
12. //定义一个数组 a, 每个元素是 Student 类型
13. Student a[100];
14. int main()
15. {
16.     int n;
17.     cin >> n;
18.     //对结构体中成员的赋值、取值。
19.     for (int i=0; i<n; i++)
20.     {
21.         cin >> a[i].name >> a[i].chinese >> a[i].math;
22.         a[i].total = a[i].chinese + a[i].math;
23.     }
24.     //冒泡排序
25.     for (int i=n-1; i>0; i--)
26.     {
27.         for (int j=0; j<i; j++)
28.         {
29.             if (a[j].total < a[j+1].total)
30.             {
31.                 swap(a[j], a[j+1]);
32.             }
33.         }
34.     }
35.     //输出
36.     for (int i=0; i<n; i++)
37.     {
38.         cout << a[i].name << ' ' << a[i].chinese << ' ' << a[i].math << ' ' <<
          a[i].total << endl;
39.     }
40.     return 0;
41. }
```