

第7章 结构体和文件

第76课 两个对象比较大小

《信息学奥赛一本通·编程启蒙 C++版》

【例 76.1】 病人排队

【题目描述】

病人登记看病，编写一个程序，将登记的病人按照以下原则排出看病的先后顺序：

老年人（年龄 ≥ 60 岁）比非老年人优先看病。

老年人按年龄从大到小的顺序看病，年龄相同的按登记的先后顺序排序。

非老年人按登记的先后顺序看病。

【输入格式】

第 1 行，输入一个小于 100 的正整数，表示病人的个数；
后面按照病人登记的先后顺序，每行输入一个病人的信息，包括：一个长度小于 1010 的字符串表示病人的 ID（每个病人的 ID 各不相同且只含数字和字母），一个整数表示病人的年龄（不超过 100 岁），中间用单个空格隔开）

【输出格式】

安排好的看病顺序输出病人的 ID，每行一个

【样例输入】

```
5
021075 40
004003 15
010158 67
021033 75
102012 30
```

【样例输出】

```
021033
010158
021075
004003
102012
```

【代码实现 1】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. struct node{
5.     string id;
6.     int age;
7.     int t;
8.     bool operator< (const node &x)const{
9.         if(age>=60&& x.age>=60){
10.             if(age==x.age) return t<x.t;
11.             else return age>x.age;
12.         }else if(age<60&& x.age<60)return t<x.t;
13.         else return age>x.age;
14.     }
15.};
16.node a[105];
17.int main(){
18.    cin>>n;
19.    for(int i=1;i<=n;i++){
20.        cin>>a[i].id>>a[i].age;
21.        a[i].t=i;
22.    }
23.    sort(a+1,a+n+1);
24.    for(int i=1;i<=n;i++) cout<<a[i].id<<"\n";
25.    return 0;
26.}
```

【代码实现 2】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. struct node{
5.     string id;
6.     int age;
7.     int t;
8. };
9. node a[105];
10. bool cmp(node x,node y){
11.     if(x.age>=60&&y.age>=60){
12.         if(x.age==y.age) return x.t<y.t;
13.         else return x.age>y.age;
14.     }else if(x.age<60&&y.age<60){
15.         return x.t<y.t;
16.     }else return x.age>y.age;
17. }
18. int main(){
19.     cin>>n;
20.     for(int i=1;i<=n;i++){
21.         cin>>a[i].id>>a[i].age;
22.         a[i].t=i;
23.     }
24.     sort(a+1,a+n+1,cmp);
25.     for(int i=1;i<=n;i++) cout<<a[i].id<<"\n";
26.     return 0;
27. }
```

练 76.1 生日排序

【题目描述】

开学了，老师要统计班里每个人的生日，并按照出生日期从早到晚排序。

【输入格式】

第一行一个整数 n ($1 \leq n \leq 100$)，班级班级的人数。

接下来 n 行，每行包含一个字符串 s 和三个整数 y, m, d ，表示姓名为 s 的同学出生日期是 y 年 m 月 d 日。

保证所有日期合法，姓名由小写字母构成，不超过 20 个字符

【输出格式】

输出 n 行，每行一个字符串表示姓名。如果有两个同学出生日期相同，输入靠后的同学先输出

【样例输入】

```
3
qwb 1996 6 30
gyt 1995 7 28
wc 1996 6 30
```

【样例输出】

```
gyt
wc
qwb
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. struct s{
5.     string name;
6.     int y,m,d,t;
7.     bool operator<(s b) const{
8.         return y<b.y||y==b.y&& m<b.m||y==b.y&&m==b.m&&d<b.d||y
           ==b.y&&m==b.m&&d==b.d&&t>b.t;
9.     }
10.} a[105];
11.int main(){
12.    cin>>n;
13.    for(int i=1;i<=n;i++){
14.        cin>>a[i].name>>a[i].y>>a[i].m>>a[i].d;
15.        a[i].t=i;
16.    }
17.    sort(a+1,a+n+1);
18.    for(int i=1;i<=n;i++) cout<<a[i].name<<endl;
19.    return 0;
20.}
```

练 76.2 精挑细选

【题目描述】

小王是公司的仓库管理员，一天，他接到了这样一个任务：从仓库中找出一根钢管。这听起来不算什么，但是这根钢管的要求可真是让他犯难了，要求如下：

这根钢管一定要是仓库中最长的；

这根钢管一定要是最长的钢管中最细的；

这根钢管一定要是符合前两条的钢管中编码最大的（每根钢管都有一个互不相同的编码，越大表示生产日期越近）。

相关的资料到是有，可是，手工从几百份钢管材料中选出符合要求的那根……要不，还是请你编写个程序来帮他解决这个问题吧。

【输入格式】

文件第一行为一个整数 n ($1 \leq n \leq 1000$)，表示仓库中所有钢管的数量。
之后 n 行，每行三个整数，分别表示一根钢管的长度（以毫米为单位，在 $[1, 10000]$ 范围内）、直径（以毫米为单位，在 $[1, 100]$ 范围内）和编码（一个 9 位整数）

【输出格式】

只有一个 9 位整数，表示选出的那根钢管的编码

【样例输入】

```
4
3000 50 872198442
3000 45 752498124
2000 60 765128742
3000 45 652278122
```

【样例输出】

```
752498124
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. struct node{
5.     int l;
6.     int d;
7.     int t;
8. };
9. node a[1005],*p;
10. bool cmp(node x,node y){
11.     return x.l<y.l||
12.         x.l==y.l&& x.d>y.d||
13.         x.l==y.l&& x.d==y.d&& x.t<y.t;
14. }
15. int main(){
16.     cin>>n;
17.     for(int i=1;i<=n;i++) cin>>a[i].l>>a[i].d>>a[i].t;
18.     p=max_element(a+1,a+n+1,cmp);
19.     cout<<(*p).t;
20.     return 0;
21. }
```

练 76.3 分数线划定

【题目描述】

世博会志愿者的选拔工作正在 A 市如火如荼的进行。为了选拔最合适的人才 A 市对所有报名的选手进行了笔试，笔试分数达到面试分数线的选手方可进入面试。面试分数线根据计划录取人数的 **150%** 划定，即如果计划录取 m 名志愿者，则面试分数线为排名第 $m \times 150\%$ (向下取整) 名的选手的分数，而最终进入面试的选手为笔试成绩不低于面试分数线的所有选手。现在就请你编写程序划定面试分数线，并输出所有进入面试的选手的报名号和笔试成绩。

【输入格式】

第一行，两个整数 $n, m (5 \leq n \leq 5000, 3 \leq m \leq n)$ ，中间用一个空格隔开，其中 n 表示报名参加笔试的选手总数， m 表示计划录取的志愿者人数。输入数据保证 $m \times 150\%$ 向下取整后小于等于 n 。

第二行到第 $n+1$ 行，每行包括两个整数，中间用一个空格隔开，分别是选手的报名号 $k (1000 \leq k \leq 9999)$ 和该选手的笔试成绩 $s (1 \leq s \leq 100)$ 。数据保证选手的报名号各不相同。

【输出格式】

第一行，有 2 个整数，用一个空格隔开，第一个整数表示面试分数线；第二个整数为进入面试的选手的实际人数。

从第二行开始，每行包含 2 个整数，中间用一个空格隔开，分别表示进入面试的选手的报名号和笔试成绩，按照笔试成绩从高到低输出，如果成绩相同，则按报名号由小到大的顺序输出。

【样例输入】

```
6 3
1000 90
3239 88
2390 95
7231 84
1005 95
1001 88
```

【样例输出】

```
88 5
1005 95
2390 95
1000 90
1001 88
3239 88
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. struct s{
4.     int name,sc;
5.     bool operator<(const s t)const{
6.         if(sc>t.sc) return true;
7.         else if(sc==t.sc){
8.             if(name<t.name) return true;
9.             else return false;
10.        }
11.        else return false;
12.    }
13.} a[10005];
14.int main(){
15.    int n,m;
16.    cin>>n>>m;
17.    for(int i=1;i<=n;i++){
18.        cin>>a[i].name>>a[i].sc;
19.    }
```

```
20.    sort(a+1,a+n+1);
21.    int y=floor(1.5*m);
22.    int x=a[y].sc;
23.    cout<<x<<" ";
24.    int cnt=0;
25.    for(int i=1;i<=n;i++){
26.        if(a[i].sc>=x) cnt++;
27.        else break;
28.    }
29.    cout<<cnt<<endl;
30.    for(int i=1;i<=cnt;i++){
31.        cout<<a[i].name<<" "<<a[i].sc<<endl;
32.    }
33.    return 0;
34. }
```

谢谢！