



CS100

信息学算法入门

第五讲

课程资料下载地址

CS100信息学算法入门公布资料的固定网站
<http://pan.baidu.com/s/1jHDcMou>

请每次课前自行将资料下载到电脑

1. 大写 （网站第130题）

```
1  #include<iostream>
2  #include<string>
3  using namespace std;
4  string s;
5  int main(){
6      cin>>s;
7      for(int i=0;i<s.size();i++)
8          if(s[i]>='a'&& s[i]<='z')
9              cout<<char(s[i]-'a'+'A');
10         else
11             cout<<s[i];
12     return 0;
13 }
```

2.凯撒加密字符串（131题）

```
1  #include<iostream>
2  #include<string>
3  using namespace std;
4  string s;
5  int main(){
6      getline(cin,s);
7      for(int i=0;i<s.size();i++){
8          if(s[i]==' ') cout<<' ';
9          else cout<<char((s[i]-'a'+1)%26+'a');
10     }
11     return 0;
12 }
```

3.数字反转（网站第132题）

```
4 string s;
5 int i;
6 int main(){
7     cin>>s;
8     if(s[0]=='-') {
9         cout<<"-";
10        s.erase(0,1);
11    }
12    for(i=s.size()-1;i>=1;i--){
13        if(s[i]!='0') break;
14    }
15    for(;i>=0;i--){
16        cout<<s[i];
17    }
```

字符串排序

简单排序 – 字符串

```
1  #include<iostream>
2  #include<algorithm>
3  #include<string>
4  #define M 10
5  using namespace std;
6  string d[M]={"New York","Shanghai","Hong Kong",
7              "London","Tokyo","Paris","Singapore",
8              "Barcelona","Berlin","Beijing"};
9  int main(){
10     for(int i=0;i<M;i++) cout<<d[i]<<endl;
11     cout<<"*****\n";
12     sort(d,d+M);
13     for(int i=0;i<M;i++) cout<<d[i]<<endl;
14     return 0;
15 }
```

单词排序

输入一行单词序列，相邻单词之间由1个或多个空格间隔，请按照字典序输出这些单词，要求**重复的单词只输出一次**。（区分大小写）数据不含除字母、空格外的其他字符。最多1000个单词

样例输入

She wants to go to Peking University to study Chinese

样例输出

Chinese
Peking
She
University
go
study
to
wants


```
1  #include<iostream>
2  #include<algorithm>
3  #include<string>
4  #define M 1000
5  using namespace std;
6  string d[M];
7  int n;
8  int main(){
9      while(cin>>d[n]) n++;
10     sort(d,d+n);
11     cout<<d[0]<<endl;
12     for(int i=1;i<n;i++)
13         if(d[i]!=d[i-1]) cout<<d[i]<<endl;
14     return 0;
15 }
```

自定义排序规则

整数奇偶排序

给定10个整数的序列，要求对其重新排序。排序要求：

- 1.奇数在前，偶数在后；
- 2.奇数按从大到小排序；
- 3.偶数按从小到大排序。

样例输入

4 7 3 13 11 12 0 47 34 98

样例输出

47 13 11 7 3 0 4 12 34 98

```
1  #include<iostream>
2  #include<algorithm>
3  #define M 10
4  using namespace std;
5  int f[M];
6  int cmp(const int &x,const int &y){
7
8
9  }
10 int main(){
11     for(int i=0;i<M;i++) cin>>f[i];
12     sort(f,f+M,cmp);
13     for(int i=0;i<M;i++) cout<<f[i]<<" ";
14     return 0;
15 }
```

自定义
排序
规则

使用自定义规则排序

简单排序：步骤

第一步 引入算法库

```
#include<algorithm>
```

第二步 对数组f中前M个数字
进行自定义规则排序

```
sort(f, f+M, cmp)
```

数组名
称为f

M个数字

自定义
排序
规则

单词排序

输入一行单词序列，相邻单词之间由1个或多个空格间隔，请按照字典序输出这些单词（**小写字母开头的单词靠前，大写字母靠后**），要求重复的单词只输出一次。数据不含除字母、空格外的其他字符。最多1000个单词

样例输入

She wants to go to Peking University to study Chinese

样例输出

go
study
to
wants
Chinese
Peking
She
University

```
1  #include<iostream>
2  #include<algorithm>
3  #include<string>
4  #define M 1000
5  using namespace std;
6  string d[M];
7  int n;
8  int cmp(const string x,const string y){
9
10
11
12 }
13 int main(){
14     while(cin>>d[n]) n++;
15     sort(d,d+n,cmp);
16     cout<<d[0]<<endl;
17     for(int i=1;i<n;i++)
18         if(d[i]!=d[i-1]) cout<<d[i]<<endl;
19     return 0;
20 }
```

自定义
排序
规则

使用自定义规则排序

struct 结构体

在处理数据时，有时需要把不同意义或不同类型的几个数据视为一个整体。

C++有结构体(struct)的类型，可以把各种类型的数据放在一起。

谁考了第k名

在一次考试中，每个学生的成绩都不相同，现知道了每个学生的学号和成绩，求考第k名学生的学号和成绩。输入第一行有两个整数，分别是学生的人数n（ $1 \leq n \leq 100$ ），和求第k名学生的k（ $1 \leq k \leq n$ ）。其后有n行数据，每行包括一个学号（整数）和一个成绩（浮点数），中间用一个空格分隔。 输出第k名学生的学号和成绩，中间用空格分隔。

样例输入

```
5 3
90788001 67.8
90788002 90.3
90788003 61
90788004 68.4
90788005 73.9
```

样例输出

```
90788004 68.4
```

```
7 struct person{  
8     int id;  
9     double s;  
10 };  
11 person p[M];
```

自定义
类型

```
12 int cmp(const person&x, const person&y){  
13     if(x.s > y.s) return 1;  
14     else return 0;  
15 }
```

自定义
排序
规则

分数排序

将n个学生的分数进行从高到低的排序，**如果同分请按照输入顺序排序。**

输入第一行为n，之后每行是学生姓名和整数分数。

样例输入

5

John 59

Bob 59

Leo 60

Wang 100

Tom 100

样例输出

Wang Tom Leo John Bob

```
7 struct person{  
8     string name;  
9     int id,s;  
10 };  
11 person p[M];  
12 int cmp(const person&x,const person&y){  
13     if(x.s>y.s) return 1;  
14     if(x.s<y.s) return 0;  
15     if(x.id<y.id) return 1;  
16     else return 0;  
17 }
```

合影效果

小云和朋友们去爬香山，为美丽的景色所陶醉，想合影留念。如果他们站成一排，男生全部在左（从拍照者的角度），并按照从矮到高的顺序从左到右排，女生全部在右，并按照从高到矮的顺序从左到右排，请问他们合影的效果是什么样的（所有人的身高都不同）？

样例输入

6

male 1.72

male 1.78

female 1.61

male 1.65

female 1.70

female 1.56

样例输出

1.65 1.72 1.78 1.70 1.61 1.56

```
7 struct person{  
8     string s;  
9     double h;  
10 };  
11 person p[M];
```

自定义
类型

```
12 int cmp(const person &x, const person &y){
13     if(x.s[0]=='m') {
14         if(y.s[0]=='f' || y.s[0]=='m' && x.h < y.h) return 1;
15         else return 0;
16     } else if(x.s[0]=='f'){
17         if(y.s[0]=='f' && x.h > y.h) return 1;
18         else return 0;
19     }
20 }
```

自定义
排序
规则

病人排队

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：

1. 老年人（年龄 ≥ 60 岁）比非老年人优先看病。
2. 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。
3. 非老年人**按登记的先后顺序**看病。

第1行表示病人个数；后面按照病人登记的先后顺序，每行输入一个病人的信息，包括病人的ID, 一个整数表示病人的年龄，中间用单个空格隔开。

样例输入

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

样例输出

```
021033
010158
021075
004003
102012
```

如何设计
struct和
cmp比较函数

课外排序练习

成绩排序

给出班里某门课程的成绩单，请你按成绩从高到低对成绩单排序输出，如果有相同分数则名字字典序小的在前。输入第一行为n ($0 < n < 20$)，表示班里的学生数目；接下来的n行，每行为每个学生的名字和他的成绩，中间用单个空格隔开。成绩为一个不大于100的非负整数。把成绩单按分数从高到低的顺序进行排序并输出，每行包含名字和分数两项，之间有一个空格。

样例输入

4

Kitty 80

Hanmeimei 90

Joey 92

Tim 28

样例输出

Joey 92

Hanmeimei 90

Kitty 80

Tim 28

奖学金

某小学最近得到了一笔赞助，打算拿出其中一部分为学习成绩优秀的前5名学生发奖学金。期末，每个学生都有3门课的成绩:语文、数学、英语。先按总分从高到低排序，如果两个同学总分相同，再按语文成绩从高到低排序，如果两个同学总分和语文成绩都相同，那么规定学号小的同学排在前面，这样，每个学生的排序是唯一确定的。

样例输入

```
6
90 67 80
87 66 91
78 89 91
88 99 77
67 89 64
78 89 98
```

样例输出

```
6 265
4 264
3 258
2 244
1 237
```

分数线划定

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。

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
```

作业

作业如何提交

<http://120.132.20.20:8080/thrall-web/main#home>