

第五章 数的存储与组织

第 5 5 课 自定义规则的排序

《信息学奥赛一本通·编程启蒙 C++版》

一、函数作为另一个函数的参数

要将函数名作为参数，需要使用函数指针，其实是函数的地址了。函数指针的定义格式为：

```
ret_type (*var_name)(arg_list);
```

表示返回值为 `ret_type`，参数列表为 `arg_list` 的函数指针 `var_name`。

例如

```
bool (*p)(int, int);
```

表示返回值为 `bool`，参数为两个 `int` 型的函数指针 `p`。

以函数指针作为形参，即可实现函数名作为参数，由另一个函数调用。

二、greater 和 less

`greater` 和 `less` 包含在头文件 `include<functional>` 中，两个对象可以被用在用在标准算法里面，如：`sort`，`merge` 或 `lower_bound`。

我们利用 `greater` 这个类我们可以降序排列一个数据容器，省得我们自己去写 `compare` 函数。例如

```
1. sort(a,a+size,greater<int>());
```

【例 55.1】 整数奇偶排序

【题目描述】

现给了你一个 **10** 个整数的序列，要求对其重新排序。排序要求：

奇数在前，偶数在后；

奇数按从大到小排序；

偶数按从小到大排序出。

【输入格式】

输入一行，包含 **10** 个整数，彼此以一个空格分开，每个整数的范围是大于等于 **0**，小于等于 **100**。

【输出格式】

按照要求排序后输出一行，包含排序后的 **10** 个整数，数与数之间以一个空格分开。

【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

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a[15],b[15],x;
4. int main(){
5.     for(int i=1;i<=10;i++){
6.         cin>>x;
7.         if(x%2) a[++a[0]]=x;
8.         else b[++b[0]]=x;
9.     }
10.    sort(a+1,a+a[0]+1,greater<int>());
11.    sort(b+1,b+b[0]+1);
12.    for(int i=1;i<=a[0];i++) cout<<a[i]<<' ';
13.    for(int i=1;i<=b[0];i++) cout<<b[i]<<' ';
14.    return 0;
15.}
```

【例 55.2】 约翰书架

【题目描述】

约翰最近买了一个书架用来存放奶牛养殖书籍，但书架很快被存满了，只剩最顶层有空余。

约翰共有 N 头奶牛 ($1 < N < 20,000$)，每头奶牛有自己的高度 H_i ， $(1 < H_i < 10000)$ N 头奶牛的总高度为 S 。书架高度为 B ($1 < B < S < 20000000007$)。

为了到达书架顶层，奶牛可以踩着其他奶牛的背，像叠罗汉一样，直到他们的总高度不低于书架高度。当然若奶牛越多则危险性越大。为了帮助约翰到达书架顶层，找出使用奶牛数目最少的解决方案吧。

【输入格式】

第 1 行：空格隔开的整数 N 和 B ；

第 2~ $N+1$ 行：第 $i+1$ 行为整数 H_i ；。

【输出格式】

能达到书架高度所使用奶牛的最少数目。

【样例输入】

6 40

6

18

11

13

19

11

【样例输出】

3

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,a[100000],sum,cnt;
4. bool cmp(int x,int y){
5.     return x>y;
6. }
7. int main(){
8.     cin>>n>>sum;
9.     for(int i=1;i<=n;i++)
10.         cin>>a[i];
11.     sort(a+1,a+1+n,cmp);
12.     for(int i=1;i<=n;i++){
13.         sum-=a[i];
14.         cnt++;
15.         if(sum<=0) break;
16.     }
17.     cout<<cnt;
18.     return 0;
19.}
```

【例 55.3】 绝对值排序

【题目描述】

输入 n ($n \leq 100$) 个整数，按照绝对值从大到小排序后输出。题目保证所有的数的绝对值都不相等。

【输入格式】

第一个数字为 n ，接着是 n 个整数。

【输出格式】

输出这 n 个整数排序后的结果。

【样例输入】

3

3 -4 2

【样例输出】

-4 3 2

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a[110],n;
4. bool cmp(int x,int y)
5. {
6.     return abs(x)>abs(y);
7. }
8. int main()
9. {
10.     cin>>n;
11.     for(int i=1;i<=n;i++)
12.         cin>>a[i];
13.     sort(a+1,a+1+n,cmp);
14.     for(int i=1;i<=n;i++)
15.         cout<<a[i]<<' ';
16.     return 0;
17.}
```

练 55.1 合影效果

【题目描述】

小严和朋友们去爬香山，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）？

【输入格式】

第一行是人数 n ($2 \leq n \leq 40$ ，且至少有 1 个男生和 1 个女生)。

后面紧跟 n 行，每行输入一个人的性别（男 **male** 或女 **female**）和身高（范围在 $[0, 2]$ 内的浮点数，单位米），两个数据之间以空格分隔。

【输出格式】

n 个浮点数，模拟站好队后，拍照者眼中从左到右每个人的身高。每个浮点数需保留到小数点后 2 位，相邻两个数之间用单个空格隔开。

【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
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. double a[45],b[45],n,x;
4. int cnta,cntb;
5. string s;
6. int main(){
7.     cin>>n;
8.     for(int i=1;i<=n;i++){
9.         cin>>s>>x;
10.        if(s=="female") a[++cnta]=x;
11.        else b[++cntb]=x;
12.    }
13.    sort(a+1,a+cnta+1,greater<double>());
14.    sort(b+1,b+cntb+1);
15.    for(int i=1;i<=cntb;i++) printf("%.2lf ",b[i]);
16.    for(int i=1;i<=cnta;i++) printf("%.2lf ",a[i]);
17.    return 0;
18.}
```

练 55.2 跳绳比赛(趣味编程)

【题目描述】

风之巔小学举行 1 分钟跳绳比赛，5 人一小组，试编一程序，输入小组内同学的跳绳次数，按次数有多到少的顺序输出

【输入格式】

输入 5 个数。

【输出格式】

次数有多到少的顺序输出这 5 个数。

【样例输入】

126 80 204 158 98

【样例输出】

204 158 126 98 80

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int main(){
4.     int i,a[6],j,t;
5.     //cout<<"输入 5 个整数: "<<endl;
6.     for (i=1;i<=5;i++){
7.         cin>>a[i];
8.     }
9.     for (int i=1;i<=4;i++){
10.        for(j=1+i;j<=5;j++){
11.            if (a[j]>a[i]){ //大小判断
12.                t=a[i]; //交换
13.                a[i]=a[j];
14.                a[j]=t;
15.            }
16.        }
17.    }
18.    for (i=1;i<=5;i++){ //输出
19.        cout<<a[i]<<" ";
20.    }
21.    return 0;
22.}
```

练 55.3 收益最大

【题目描述】

农夫 John 余下了 m 批干草无法处理,他准备要开一个拍卖会去出售他的干草。现在有 n 个顾客,每个顾客的报价是 a_i 。现在 John 要确定一个单价,所有报价大于等于单价的顾客将会买到 1 批干草(m 批干草不用全卖完),总共获得的金钱作为收益。那么问题来了,如何设定单价,使得收益最大。

【输入格式】

第一行两个整数 m, n , 分别表示 m 批干草和 n 个顾客。第二行 n 个整数, a_i 表示第 i 个顾客的报价。

数据范围: $1 < n, m \leq 1000, 1 < a_i \leq 10000$ 。

【输出格式】

两个用空格分隔的整数,分别表示单价和总收益。如果有多个相等的最大收益,选取单价最小的那个。

【样例输入】

```
5 4
2 8 10 7
```

【样例输出】

```
7 21
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int grass,n,a[1004],ans,cnt,prize;
4. int main(){
5.     cin>>grass>>n;
6.     for(int i=1;i<=n;i++)cin>>a[i];
7.     sort(a+1,a+n+1,greater<int>());
8.     for(int i=1;i<=n;i++){
9.         cnt=min(i,grass);
10.        if(ans<=cnt*a[i]){
11.            ans=cnt*a[i];
12.            prize=a[i];
13.        }
14.    }
15.    cout<<prize<<' '<<ans;
16.    return 0;
17.}
```


练 55.4 沙堡

【题目描述】

约翰用沙子建了一座城堡。正如所有城堡的城墙，这城墙也有许多枪眼，两个相邻枪眼中间那部分叫作“城齿”。城墙上一共有 N ($1 < N \leq 25000$) 个城齿，每一个都有一个高度 M_i ($1 \leq M_i \leq 10^5$)。

现在约翰想把城齿的高度调成某种顺序下的 B_i ($1 \leq B_i \leq 10^5$)。一个城齿每提高一个单位的高度，约翰需要 X ($1 \leq X \leq 100$) 元；每降低一个单位的高度，约翰需要 Y ($1 \leq Y \leq 100$) 元，问约翰最少可用多少钱达到目的。

【输入格式】

第一行三个整数 N, X, Y 。

接下来 N 行，每行两个整数 M_i, B_i

【输出格式】

一个整数，表示最少的花费。

【样例输入】

3 6 5

3 1

1 2

1 2

【样例输出】

11

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. long long n,x,y,a[25005],b[25005],sum;
4. int main(){
5.     cin>>n>>x>>y;
6.     for(int i=0;i<n;i++)cin>>a[i]>>b[i];
7.     sort(a,a+n);
8.     sort(b,b+n);
9.     for(int i=0;i<n;i++){
10.         if(a[i]>b[i])sum+=(a[i]-b[i])*y;
11.         if(a[i]<b[i])sum+=(b[i]-a[i])*x;
12.     }
13.     cout<<sum;
14.     return 0;
15.}
```

谢谢！

