

《插入排序过程》

```
5 | cin>>n;  
6 | for(int i=1;i<=n;i++){  
7 |     cin>>a[i];  
8 | }  
9 | for(int i=2;i<=n;i++){  
10 |     for(int j=i-1;j>=1;j--){  
11 |         if(a[j]>a[j+1]){  
12 |             swap(a[j],a[j+1]);  
13 |         }else{  
14 |             break;  
15 |         }  
16 |     }  
17 |     for(int j=1;j<=n;j++){  
18 |         cout<<a[j]<<" ";  
19 |     }  
20 |     cout<<endl;  
21 | }  
22 | return a;
```

《新年礼物》

```
16     int n,Max;
17     cin>>Max>>n;
18     for(int i=1;i<=n;i++){
19         cin>>a[i];
20     }
21     csort(n);
22     int ans=0,r=n,l=1;
23     while(l<=r){
24         if(a[l]+a[r]<=Max){
25             l++,r--,ans++;
26         }else{
27             r--,ans++;
28         }
29     }
30     cout<<ans;
31     return 0;
```

《总统选举》

```
16     int n;  
17     cin>>n;  
18     for(int i=1;i<=n;i++){  
19         cin>>a[i];  
20     }  
21     csort(n);  
22     n=n/2+1;  
23     int sum=0;  
24     for(int i=1;i<=n;i++){  
25         sum+=(a[i]/2+1);  
26     }  
27     cout<<sum;  
28     return 0;  
29 }
```

```
3 int a[10000];
4 int main() { // 冒泡排序过程
5     int n;
6     cin>>n;
7     for(int i = 0; i < n; i++)
8         cin>>a[i];
9     for(int i=0; i<n-1; i++) {
10         for(int j=0; j<n-i-1; j++) {
11             if(a[j] > a[j+1]) {
12                 swap(a[j],a[j+1]);
13             }
14             for(int k=0; k<n; k++) {
15                 cout<<a[k]<<" ";
16             }
17             cout<<endl;
18         }
19     }
20     return 0;
```

```
3  int a[100005];
4  int bfind(int x,int L,int R){
5      while(L<R){
6          int mid=(L+R)>>1;
7          if(a[mid]>=x){//满足条件,
8              R=mid; //但要最小
9          }else{
10             L=mid+1;
11         }
12     }
13     return L;
14 }
15 int main() {
16     int n,m,x;
17     cin>>n>>m;
18     for(int i=1;i<=n;i++){
19         cin>>a[i];
20     }
21     while(m--){
22         cin>>x;
23         int k=bfind(x,1,n);
24         if(a[k]!=x)cout<<"-1"<<endl;
25         else cout<<k-1<<endl;
26     }
27     return 0;
```



实验舱
青少年编程
走近科学 走进名校

2022暑期蛟龙二班 day9

Vector 容器应用

zlj

2022.7

稳定排序与不稳定排序

稳定：插入、冒泡 、归并

不稳定：选择

Vector（向量）：

是一种顺序容器。（相当于数组）特点是vector正好弥补了静态数组缺陷：随机访问快，在中间插入和删除慢，但在末端插入和删除快。按需扩张收缩。



1、Vector的定义与初始化

定义：

```
vector<int> a
```

Vector 遍历：

1、 **vector<int>::iterator it;** //定义一个迭代器

2、Vector支持 []运算。

a[2]=6 可像数组一样按下标访问

1、Vector定义:

```
vector<int>v1;           //vector元素为 int 型  
vector<string>v2;       // vector元素为string型
```

初始化

```
vector<int > a(4,10)
```

定义一个容器 a , 前4个值为 都为10

```
vector<int > b={1,2,3};
```

定义一个容器 b , 前3个值为 1,2,3

Vector 遍历:

1、 **vector<int>::iterator it;** //定义一个迭代器

2、 Vector支持 []运算。

a[2]=6 可像数组一样按下标访问

//遍历数据

```
for(int i=0;i<b.size();i++){  
    cout<<b[i]<<' ';<br>  
}
```

Vector 遍历:

```
22     vector<int>::iterator it;  
23     for( it=b.begin();it!=b.end();it++){  
24         cout<<*it<<' ';  
25     }
```

2、**vector** 基本函数操作

常用函数操作汇总:

`v1.push_back()` //在数组的最后添加一个数据
`v1.pop_back()` //去掉数组的最后一个数据
`v1.front()` `v1.back()` //返回第一个元素(最后一个)
`v1.begin()` //得到数组头的指针, 用迭代器接受
`v1.end()` //得到数组的最后一个单元+1的指针, 用迭代器接受
`v1.clear()` // 移除容器中所有数据
`v1.empty()` //判断容器是否为空
`v1.erase(pos)` //删除pos位置的数据
`v1.erase(beg,end)` // 删除[beg,end)区间的数据
`v1.size()` //返回容器中实际数据的个数
`v1.insert(pos,data)` //在pos处插入数据

`reverse(v1.begin(),v1.end())` //将v1容器数据倒置。

`find(v.begin(),v.end(),3)` //查找V容器里3, 有返回迭 代器, 没有返回 `end()`

2、vector 的应用

分类、排序、去重
跟数组类似

```

3 int a[5000005];
4 int check(int x) { //x=123 3 2 1
5     int ji=1;
6     while(x!=0) {
7         ji*=x%10;
8         x/=10;
9     }
10    return ji;
11 }
12 int main() {
13     int n,x,cnt=0,Max=-1,cj=0x3f3f3f3f;
14     cin>>n;
15     for(int i=1; i<=n; i++) {
16         cin>>x;
17         int k=check(x);
18         if( a[k]==0 ){
19             cnt++;
20         }
21         a[k]++; //标记
22         if( a[k]>Max || (a[k]==Max)&&(k<cj) ){ //打擂台 找规模最大同时乘积最小
23             Max=a[k],cj=k;
24         }
25     }
26     cout<<cnt<<" "<<cj<<endl;
27     return 0;
28 }

```

练习1:《统计》

输入n个整数，将其中的奇数和偶数分别显示出来（ $1 < n < 30$ ）

输入

第一行：一个整数n。第二行：n个空格隔开的整数。

输出

第一行：若干个奇数。第二行：若干个偶数。（每个数前面都有一个空格）

样例输入

10

21 12 33 43 59 68 77 18 19 40

样例输出

21 33 43 59 77 19

12 68 18 40

练习2 《倒序输出》

输入若干个数（一行，两数间用空格分隔），并将它们存入动态数组**a** 中，倒序输出这些数。

```
using namespace std;
int main() {
    vector<int>a;
    vector<int>::iterator it;
    int x;
    while(cin>>x) {
        a.push_back(x);
    }
    ✗ reverse(a.begin(),a.end());
    for(it=a.begin(); it!=a.end(); it++)
        cout<<*it<<" ";
    cout<<endl;
    return 0;
}
```

Sort 排序

```
sort( a.begin(), a.end() )
```

练习4： 《整数去重》

输入若干个整数，去掉重复的数字，（只保留1个），并按升序输出；

样例：

输入：

4 2 4 1 9 1

输出：

1 2 4 9

```
3 vector<int>a;
4 int main() {
5     int x;
6     while(cin>>x){
7         a.push_back(x);
8     }
9     sort(a.begin(),a.end());
10    for(int i=0;i<a.size();i++){
11        if(a[i]!=a[i+1]){
12            cout<<a[i]<<' ';
13        }
14    }
15    return 0;
```



```
17 while(cin>>x){
18     if(a.size()==0|| find(a.begin(),a.end(),x)==a.end() ){
19         a.push_back(x);
20     }
21 }
22 sort(a.begin(),a.end());
23 for(int i=0;i<a.size();i++){
24     cout<<a[i]<<" ";
25 }
```

练习： 《前缀串》

3、Vector的Insert操作

- Insert(x, y)是Vector的成员函数，其中，x是一个迭代器，y是一个具体的值。
- Insert(x, y)在x对应的元素之前插入了一个值为y的元素。

Insert.cpp

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int main() {
4      vector<int> a = {0, 1, 2};
5      for (unsigned i = 0; i < a.size(); i++)
6          cout << a[i] << ' ';
7      cout << endl;
8      a.insert(a.begin() + 1, 3);
9      for (unsigned i = 0; i < a.size(); i++)
10         cout << a[i] << ' ';
11     cout << endl;
12     return 0;
13 }
```

D:\XYX\Programming\Programming\2017-08-

```
0 1 2
0 3 1 2
```

```
-----
Process exited after 0.03892 seconds
请按任意键继续. . .
```

4、Vector的Erase操作

- `Erase(x)`, `Erase(x,y)`是Vector的成员函数，其中，`x,y`是迭代器。
- 分别能够删除`x`处的元素或区间`[x,y)`内的元素。
- 由于Vector的分块存储方式，`Erase`的复杂度为 $O(\text{Log}(\text{size()}))$ 。

Delete.cpp

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int main() {
4      vector<int> a = {1, 2, 3};
5      a.erase(a.begin() + 1);
6      for (unsigned i = 0; i < a.size(); i++)
7          cout << a[i] << ' ';
8      cout << endl;
9      vector<int> b = {1, 2, 3, 4, 5};
10     b.erase(b.begin() + 1, b.begin() + 4);
11     for (unsigned i = 0; i < b.size(); i++)
12         cout << b[i] << ' ';
13     cout << endl;
14     return 0;
15 }
```

D:\XYX\Programming\Programming\2017-08-

```
1 3
1 5
```

Process exited after 0.07191 seconds
请按任意键继续. . .

vector 的成员函数表

函数名	功能	复杂度
size()	返回数组的 元素数	$O(1)$
push_back()	在向量末尾添加元素x	$O(1)$
pop_back()	删除最后一个元素	$O(1)$
begin()	返回数组开头的迭代器	$O(1)$
end()	最后一个元素的后一个位置的迭代器	$O(1)$
insert(p,x)	在位置p处插入元素x	$O(n)$
erase(p)	删除位置p 的元素	$O(n)$
clear()	删除数组中所有元素	$O(n)$

4、find (区间, 值) 返回迭代器

```
27     vector<int>::iterator it;  
28     it=find(b.begin(),b.end(),18);  
29     if(it!=b.end()){  
30         cout<<"yes";  
31     }else{  
32         cout<<"no";  
33     }
```


二班内容复习：

- 1、一维数组
- 2、字符数组
- 3、string
- 4、函数
- 5、进制转换
- 6、二分查找
- 7、vector