

第五章 数的存储与组织

第 5 1 课 认识 `vector`

《信息学奥赛一本通·编程启蒙 C++版》

一、vector介绍

vector是标准库中常见的一种动态序列容器容器，表示可变大小数组的序列容器，可以用来代替c++原本的数组。

就像数组一样，vector也采用的连续存储空间来存储元素。也就是意味着可以采用下标对vector的元素进行访问，和数组一样高效。数组就像是一个线段，不能延伸，vector就像是射线，空间不够用的时候可以往后延伸，它的大小是可以动态改变的，并且它的大小会被器自动处理。

二、vector 的几种常用语法

限于篇幅，下面罗列了vector的几种常用的语法，读者如果想要获得更多的关于vector的用法，可以打开网页：<https://cplusplus.com>，然后搜索vector。

语法	说明
<code>vector<int> v;</code>	创建空 vector
<code>v.push_back(x)</code>	向尾部增加一个元素 x
<code>v.insert(pos, x)</code>	向 pos 地址指向元素前增加一个元素 x
<code>v[i]</code>	访问 i 位置元素
<code>v.pop_back()</code>	删除向量中最后一个元素
<code>v.clear()</code>	清空向量中所有元素
<code>v.empty()</code>	判断向量是否为空
<code>v.size()</code>	返回向量中元素的个数
<code>v.begin()</code>	返回向量头指针（迭代器），指向第一个元素
<code>v.end()</code>	返回向量尾指针（迭代器），指向最后一个元素+1 位置
<code>v.erase(v.begin()+i)</code>	删除第 i 位位置的元素

【例 51.1】 删除元素

【题目描述】

把一个数组的第 x 个位置的元素删除掉。

【输入格式】

有三行

第一行有一个整数 n ($n \leq 10$)

第二行有 n 个整数

第三行有一个整数 x ，为要删除的位置。

【输出格式】

更新后的数组。

【样例输入】

5

1 2 3 4 5

3

【样例输出】

1 2 4 5

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,x,y;
4. vector<int> v;
5. int main(){
6.     cin>>n;
7.     for(int i=1;i<=n;i++) {
8.         cin>>x;
9.         v.push_back(x);
10.    }
11.    cin>>x;
12.    v.erase(v.begin()+x-1);
13.    for(int i=0;i<n-1;i++)
14.        cout<<v[i]<<' ';
15.    return 0;
16.}
```

【例 51.2】 插入元素

【题目描述】

在一个数组的第 x 个位置插入一个新的数 y 。

【输入格式】

有四行

第一行有一个整数 n ($5 \leq n \leq 10$)

第二行有 n 个整数

第三行有一个整数 x ，为要插入的位置 第四行有一个整数 y ，为要插入的整数。

【输出格式】

更新后的数组。

【样例输入】

5

7 2 3 4 5

2

9

【样例输出】

7 9 2 3 4 5

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,x,y;
4. vector<int> v;
5. int main(){
6.     cin>>n;
7.     for(int i=1;i<=n;i++) {
8.         cin>>x;
9.         v.push_back(x);
10.    }
11.    cin>>x>>y;
12.    v.insert(v.begin()+x-1,y);
13.    for(int i=0;i<=n;i++)
14.        cout<<v[i]<<' ';
15.    return 0;
16.}
```

【例 51.3】 平移数据

【题目描述】

将 **a** 数组中第一个元素移到数组末尾,其余数据依次往前平移一个位置。

【输入格式】

第一行为数组 **a** 的元素个数;

第二行为 **n** 个小于 **1000** 的正整数。

【输出格式】

平移后的数组元素,每个数用一个空格隔开。

【样例输入】

10

1 2 3 4 5 6 7 8 9 10

【样例输出】

2 3 4 5 6 7 8 9 10 1

【代码实现 1】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,x;
4. vector<int> myvector;
5. int main(){
6.     cin>>n;
7.     for (int i=1; i<=n; i++) {
8.         cin>>x;
9.         myvector.push_back(x);
10.    }
11.    x=myvector[0];
12.    myvector.push_back(x);
13.    myvector.erase (myvector.begin()+0);
14.    for (unsigned i=0; i<myvector.size(); ++i)
15.        cout<< myvector[i] << ' ';
16.    return 0;
17. }
```

【代码实现 2】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,x,y;
4. int main(){
5.     cin>>n;
6.     cin>>y;
7.     for(int i=2;i<=n;i++) {
8.         cin>>x;
9.         cout<<x<<' ';}
10.    cout<<y;
11.    return 0;
12. }
```

练 51.1 向量点积计算

【题目描述】

在线性代数、计算几何中，向量点积是一种十分重要的运算。

给定两个 n 维向量， $\vec{a} = (a_1, a_2, a_3, \dots, a_n)$ 和 $\vec{b} = (b_1, b_2, b_3, \dots, b_n)$ ，求点积
 $\vec{a} \cdot \vec{b} = a_1b_1 + a_2b_2 + a_3b_3 + \dots + a_nb_n$ 。

【输入格式】

第一行是一个整数 n ， $1 \leq n \leq 1000$ 。

第二行包含 n 个整数

第三行包含 n 个整数

相邻整数之间用单个空格隔开。每个整数的绝对值都不超过 1000。

【输出格式】

输出该学号成绩

【样例输入】

7

7 8 9 4 8 7 8

3 7 4 7 9 9 7

【样例输出】

332

【代码实现】

```
1. #include <bits/stdc++.h>
2. using namespace std;
3. int n, sum = 0, a[1000], b[1000];
4. int main() {
5.     cin >> n;
6.     for (int i = 0; i < n; i++) cin >> a[i];
7.     for (int i = 0; i < n; i++) cin >> b[i];
8.     for (int i = 0; i < n; i++) sum += a[i] * b[i];
9.     cout << sum << endl;
10.    return 0;
11.}
```

练 51.2 老鹰捉小鸡(趣味编程)

【题目描述】

狐狸老师和格莱尔等 5 位小朋友玩老鹰捉小鸡的游戏，狐狸老师当老鹰，排在第一位的小朋友当“母鸡”，其他 4 位小朋友当“小鸡”。但是“母鸡”很辛苦，所以过一段时间“母鸡”需要排到队伍最后成为“小鸡”，让第二位小朋友当“母鸡” 试编一程序，模拟 10 次位置变化的过程。

【输入格式】

无。

【输出格式】

每行输出每一次的位置

【样例输入】

无

【样例输出】

```
1: 1 2 3 4 5
2: 2 3 4 5 1
3: 3 4 5 1 2
4: 4 5 1 2 3
5: 5 1 2 3 4
6: 1 2 3 4 5
7: 2 3 4 5 1
8: 3 4 5 1 2
9: 4 5 1 2 3
10: 5 1 2 3 4
```

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int main(){
4.     int i,j,a[6],n;
5.     for (int i=1;i<=6;i++){
6.         a[i]=i;
7.     }//输出第一次的位置
8.     i=1;
9.     cout<<i<<": ";
10.    for (int j=1;j<6;j++){
11.        cout<<a[j]<<" ";
12.    }
13.    cout<<endl;
14.    for (int i=2;i<=10;i++){
15.        //移动位置
16.        for (int j=0;j<=4;j++){
17.            a[j]=a[j+1];
18.        }
19.        a[5]=a[0];
20.        //输出位置
21.        cout<<i<<": ";
22.        for (int j=1;j<=5;j++){
23.            cout<<a[j]<<' ';
24.        }
25.        cout<<endl;
26.    }
27.    return 0;
28. }
```

练 51.3 纸杯猜数

【题目描述】

你和小华正在玩纸杯猜数游戏。

小华在桌面上扣放了一排 n 个纸杯，依次标号为 $1\sim n$ 。每个纸杯内侧写有一个数字。

小华先将每个纸杯上的数字告诉小明，然后他快速的进行了如下 m 次交换操作：

选择两个纸杯 u 和 v ，交换它们的位置。（ u, v 相同表示本次不交换）

现在小华想让你回答： m 次操作后依次排列的纸杯上的数字分别是多少。

【输入格式】

第一行输入两个数 n, m ，分别表示纸杯数、操作数（ $1 \leq n, m \leq 50000$ ） 第二行输入 n 个数，分别表示每个纸杯内侧的数字之后 m 行，每行两个数 u, v ，表示将这两个纸杯交换。

【输出格式】

输出一行 n 个数，表示交换后每个纸杯内侧的数字依次是多少。以空格隔开

【样例输入】

```
8 3
1 4 7 20 5 3 11 6
3 7
2 6
5 3
```

【样例输出】

```
1 3 5 20 11 4 7 6
```

【代码实现 1】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,m,x,y;
4. vector<int> v;
5. int main(){
6.     cin>>n>>m;
7.     for(int i=1;i<=n;i++){
8.         cin>>x;
9.         v.push_back(x);
10.    }
11.    for(int i=1;i<=m;i++){
12.        cin>>x>>y;
13.        swap(v[x-1],v[y-1]);
14.    }
15.    for (unsigned i=0; i<v.size(); ++i)
16.        cout<<v[i]<<' ';
17.    return 0;
18. }
```

【代码实现 2】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a[50005],n,m,u,v;
4. int main(){
5.     cin>>n>>m;
6.     for(int i=1;i<=n;i++) cin>>a[i];
7.     for(int i=1;i<=m;i++){
8.         cin>>u>>v;
9.         swap(a[u],a[v]);
10.    }
11.    for(int i=1;i<=n;i++)
12.        cout<<a[i]<<' ';
13.    return 0;
14. }
```

谢谢！

