

第五章 数的存储与组织

第 5.7 课 全排列函数

《信息学奥赛一本通·编程启蒙 C++版》

【例 57.1】 排列

【题目描述】

从 1 开始到 n ($1 \leq n \leq 9$) 这 n 个数字一共可以组成 $n!$ 个不同的排列。现在给出其中一个排列，求它是所有排列中字典序第几小的排列？

【输入格式】

第一行一个整数 n 。

第二行一个整数 x ，是由 1 到 n 这 n 个数字组成的某个排列。

【输出格式】

一个整数，表示该排列是字典序第几小的。

【样例输入】

4

2143

【样例输出】

8

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,cnt;
4. char c;
5. vector<int> a,b;
6. int main(){
7.     cin>>n;
8.     for(int i=1;i<=n;i++){
9.         cin>>c;b.push_back(c-48);
10.        a.push_back(i);
11.    }
12.    do{
13.        cnt++;
14.        if(a==b) break;
15.    }while(next_permutation(a.begin(),a.end()));
16.    cout<<cnt;
17.    return 0;
18.}
```

【例 57.2】 上一个排列

【题目描述】

地球人都知道，从 1 开始到 n ($1 \leq n \leq 9$) 这 n 个数字一共可以组成 $n!$ 个不同的排列。现在把这 n 个排列按字典序从小到大排序，给出其中一个排列，求该排列的上一个排列。

【输入格式】

第一行一个整数 n ，表示排列的长度。

第二行 n 个用空格分隔的整数，为 1 到 n 组成的某个排列。

【输出格式】

如果上一个排列存在，则输出；否则输出 "ERROR"

【样例输入】

4

2 1 3 4

【样例输出】

1 4 3 2

【代码实现】

```
1. #include <bits/stdc++.h>
2. using namespace std;
3. int n, a[10];
4. int main(){
5.     cin >> n;
6.     for (int i = 1; i <= n; ++i)
7.         cin >> a[i];
8.     if (prev_permutation(a + 1, a + 1 + n))
9.     {
10.         for (int i = 1; i <= n; ++i)
11.             printf("%d ", a[i]);
12.         putchar('\n');
13.     }
14.     else
15.         printf("ERROR\n");
16. return 0;
17. }
```

【例 57.3】火星人

【题目描述】

人类终于登上了火星的土地并且见到了神秘的火星人。人类和火星人都无法理解对方的语言，但是我们的科学家发明了一种用数字交流的方法。这种交流方法是这样的，首先，火星人把一个非常大的数字告诉人类科学家，科学家破解这个数字的含义后，再把一个很小的数字加到这个大数上面，把结果告诉火星人，作为人类的回答。

火星人用一种非常简单的方式来表示数字——掰手指。火星人只有一只手，但这只手上有成千上万的手指，这些手指排成一列，分别编号为 1, 2, 3……。火星人的任意两根手指都能随意交换位置，他们就是通过这方法计数的。

一个火星人用一个人类的手演示了如何用手指计数。如果把五根手指——拇指、食指、中指、无名指和小指分别编号为 1, 2, 3, 4 和 5，当它们按正常顺序排列时，形成了 5 位数 12345，当你交换无名指和小指的位置时，会形成 5 位数 12354，当你把五个手指的顺序完全颠倒时，会形成 54321，在所有能够形成的 120 个 5 位数中，12345 最小，它表示 1；12354 第二小，它表示 2；54321 最大，它表示 120。下表展示了只有 3 根手指时能够形成的 6 个 3 位数和它们代表的数字：

三进制数 123 132 213 231 312 321

代表数字 1 2 3 4 5 6

现在你有幸成为了第一个和火星人交流的地球人。一个火星人会让你看他的手指，科学家会告诉你要加上去的很小的数。你的任务是，把火星人用手指表示的数与科学家告诉你的数相加，并根据相加的结果改变火星人手指的排列顺序。输入数据保证这个结果不会超出火星人手指能表示的范围。

【输入格式】

三行，第一行有一个正整数 N ，表示火星人手指的数目 ($1 \leq N \leq 10000$)。第二行是一个正整数 M ，表示要加上的小整数 ($1 \leq M \leq 100$)。下一行是 1 到 N 这 N 个整数的一个排列，用空格隔开，表示火星人手指的排列顺序。

【输出格式】

只有一行，这一行含有 N 个整数，表示改变后的火星人手指的排列顺序。每两个相邻的数中间用一个空格分开，不能有多余的空格。

【样例输入】

```
5
3
1 2 3 4 5
```

【样例输出】

```
1 2 4 5 3
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,m,a[10010];
4. int main(){
5.     cin>>n>>m;
6.     for(int i=1;i<=n;i++)
7.         cin>>a[i];
8.     for(;m--;)
9.         next_permutation(a+1,a+1+n);
10.    for(int i=1;i<=n;i++)
11.        cout<<a[i]<<' ';
12.    return 0;
13.}
```

练 57.1 全排列问题

【题目描述】

输出自然数 1 到 n 所有不重复的排列，即 n 的全排列，要求所产生的任一数字序列中不允许出现重复的数字。

【输入格式】

一个整数 n ($1 \leq n \leq 9$)。

【输出格式】

由 $1 \sim n$ 组成的所有不重复的数字序列，每行一个序列。每个数字保留 5 个场宽。

【样例输入】

3

【样例输出】

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

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n,a[15];
4. int main(){
5.     scanf("%d",&n);
6.     for(int i=1;i<=n;i++) a[i]=i;
7.     do{
8.         for(int i=1;i<=n;i++)
9.             printf("%5d",a[i]);
10.        puts("");
11.    }
12.    while(next_permutation(a+1,a+n+1));
13.    return 0;
14.}
```

谢谢！

—