

Комбинаторика Перестановка транспозицией соседних элементов

Code by Xumingchuan

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<stdbool.h>
4  #define N 4
5
6  int A[N];
7  bool Dir[N];//false - left; true - right
8  void Init(){
9      for(int i=0;i<N;i++)
10         A[i]=i+1;
11 }
12
13 void InitDir(){
14     for(int i=0;i<N;i++)
15         Dir[i]=false;
16 }
17
18 void SwapA(int i,int j){
19     int Tmp=A[i];
20     A[i]=A[j];
21     A[j]=Tmp;
22 }
23
24 void SwapDir(int i,int j){
25     int Tmp=Dir[i];
26     Dir[i]=Dir[j];
27     Dir[j]=Tmp;
28 }
29
30 void Print(){
31     for(int i=0;i<N;i++)
32         printf("%d ",A[i]);
33     printf("\n");
34 }
35
36 bool MoveActive(){
37     int Max=1;//max number
38     int Index=-1;//position of max number
39     //find the biggest active number
40     for(int i=0;i<N;i++){
41         if(A[i]<Max) continue;
42         if( (i<N-1&&A[i]>A[i+1]&&Dir[i]) || (i>0&&A[i]>A[i-1]&&!Dir[i]) ){
43             Max=A[i];
44             Index=i;
45         }
46     }
47     if(Index==-1) return false;//no active number
48
49     //Swap [Index]==true - right exchange [Index]==false - left exchange
50     if(Dir[Index]){
51         SwapA(Index,Index+1);
52         SwapDir(Index,Index+1);
53     }
54     else{
55         SwapA(Index,Index-1);
56         SwapDir(Index,Index-1);
57     }
58
59     //Change all direction of number which bigger than biggest active number
60     for(int i=0;i<N;i++)
61         if(A[i]>Max) Dir[i]=!Dir[i];
62
63     return true;
64 }
65
66 int main(){
67     Init();
68     InitDir();
69     printf("The Permutation with exchange:\n");
70     do{
71         Print();
72     } while(MoveActive());
73
74     return 0;
75 }
```

ВЫВОД:

```
The Permutation with exchange:
1 2 3 4
1 2 4 3
1 4 2 3
4 1 2 3
4 1 3 2
1 4 3 2
1 3 4 2
1 3 2 4
3 1 2 4
3 1 4 2
3 4 1 2
4 3 1 2
4 3 2 1
3 4 2 1
3 2 4 1
3 2 1 4
2 3 1 4
2 3 4 1
2 4 3 1
4 2 3 1
4 2 1 3
2 4 1 3
2 1 4 3
2 1 3 4
```

InitDir() – Инициализируем массив Dir. Значение false в этом массиве типа bool представляет состояние активности элемента влева, а значение true представляет состояние активности элемента вправо (направление активного состояния: если значение этого разряда больше следующего разряда, активное состояние этого элемента находится вправо; Значение этого бита больше предыдущего, и этот элемент активен влева)

MoveActive() – Этот способ порождения следующей перестановки делится на три этапа:

1. Найдём самый большой элемент , если он не найден, программа закончена.
2. Заменим наибольший активный элемент и соседний элемент, находящий в его активном направлении
3. Изменим направление всех элементов которые больше, чем самый большой активный элемент

Если функция возвращает true, это означает что существует следующая перестановка, потом печатается и продолжаемся.