一、题目说明

题目是31. Next Permutation，英文太差看不懂，翻译了一下。才知道是求字典顺序下的下一个排列，不允许使用额外空间。题目难度是Medium！

二、我的实现

首先要进一步理解题目，以`1->2->3`为例，字典顺序如下：

```
（1） 1->2->3;
（2） 1->3->2;
（3） 2->1->3;
（4） 2->3->1;
（5） 3->1->2;
（6） 3->2->1;
（7） 1->2->3;
```

如何从（1)-> (2) ->(3)-> (4) ->(5)-> (6) ->(7)实现状态转换？以(3)->(4)为例：

从列表lists的最右边起，

if(lists[t] < lists[t-1]) {

  swap(lists[t-1],max{lists[t]...lists[listSize-1]})

  sort(lists[t],lists[listSize-1]);

}

从(6)->(7)，sort(lists[0],lists[listSize-1])即可。

代码如下：

```
#include<iostream>
#include<vector>
#include<algorithm>
using namespace std;

class Solution {
public:
    void nextPermutation(vector<int>& nums){
        if(nums.size()<=1) return ;
        bool flag = false;
        for(int t=nums.size()-1;t>0;t--){
            if(nums[t]>nums[t-1]){
                //find the smallest between nums[t] to nums[t-1]
                flag = true;
                int max = nums[t];
                int maxIndex = t;

                for(int k=nums.size()-1;k>=t;k--){
                    if(nums[t-1]<nums[k]){
                        max = nums[k];
                        maxIndex = k;
                        break;
                    }
                }
                int tmp = nums[t-1];
                nums[t-1] = nums[maxIndex];
                nums[maxIndex] = tmp;

                //从t..size()-1重新排序
                int len = nums.size()-t;
                for(int s=0;s<(len+1)/2;s++){
                    tmp = nums[t+s];
                    nums[t+s] = nums[nums.size()-s-1];
                    nums[nums.size()-s-1] = tmp;
                 }
                break;
            }
        }
         if(!flag){
            int tmp,len = nums.size();
            for(int t=0;t<(len+1)/2;t++){
                tmp = nums[t];
                nums[t] = nums[len-t-1];
                nums[len-t-1] = tmp;
             }
         }
    }
};
int main(){
    Solution s;
    vector<int> v;

    v = {1,3,2};
    s.nextPermutation(v);
    for(vector<int>::iterator it=v.begin();it!=v.end();it++){
        cout<<*it<<" ";
    }
    cout<<endl;

    v = {5,4,7,5,3,2};
    s.nextPermutation(v);
    for(vector<int>::iterator it=v.begin();it!=v.end();it++){
        cout<<*it<<" ";
    }
    cout<<endl;

    v = {3,2,1};
    s.nextPermutation(v);
    for(vector<int>::iterator it=v.begin();it!=v.end();it++){
        cout<<*it<<" ";
    }
    cout<<endl;

    v = {1,5,1};
    s.nextPermutation(v);
    for(vector<int>::iterator it=v.begin();it!=v.end();it++){
        cout<<*it<<" ";
    }
    cout<<endl;
    return 0;
}
```

三、改进措施

提交后，性能如下：

```
Runtime: 8 ms, faster than 78.45% of C++ online submissions for Next
Permutation.
Memory Usage: 8.6 MB, less than 88.17% of C++ online submissions for Next
Permutation.
```

差不多了，就不优化了。