89. Gray Code

题目描述：https://leetcode.com/problems/gray-code/

> 给定一个位数，求这个位数的所有Gray Code，其中上一个数和本数的位数相差1就是Code
> 例如：
>
> ```
> 给定 n = 2
>
> 00 = 0
> 01 = 1
> 11 = 3
> 10 = 2
>
> 返回[0,1,3,2]
> ```

解题思路1：

0 0 0

0 0 1

------ n = 1

0 1 1

0 1 0

------ n = 2

1 1 0

1 1 1

1 0 1

1 0 0

------ n = 3

依次 ：

n = 4 便是由 n = 3的情况 + n = 3倒着依次与8所得

代码1：

```
class Solution {
public:
    vector<int> grayCode(int n) {
        if(n == 0) {
            return {0};
        }
        vector<int> res;
        res.push_back(0);
        getCode(res, n);
        return res;
    }
    void getCode(vector<int> &res, int n) {
        if(n == 1) {
            res.push_back(1);
            return ;
        }
        getCode(res, n-1);
        int k = res.size();
        for(int i = k-1; i >= 0; i--) {
            int newNum = res[i] | k;
            res.push_back(newNum);
        }
    }
};
```

解题思路2：

按照SubSet那样求

代码2：[求的顺序不一样]

```
class Solution {
public:
    vector<int> grayCode(int n) {
        vector<int> res;
        res.push_back(0);
        getCode(res, n, 0, 0);
        return res;
    }
    bool isExist(vector<int> res, int n) {
        for(auto i: res) {
            if(i == n) {
                return true;
            }
        }
        return false;
    }
    void getCode(vector<int> &res, int n, int num, int pos) {
        if(pos == n) {
            return ;
        }
        for(int i = pos; i < n; i++) {
            int newNum = 1 << i;
            num = num ^ newNum;
            res.push_back(num);
            getCode(res, n, num, i+1);
            num = num ^ newNum;
        }
    }
};
```