

# 水中的鱼

Saturday, December 22, 2012

## [LeetCode] Gray Code 解题报告

The gray code is a binary numeral system where two successive values differ in only one bit.

Given a non-negative integer  $n$  representing the total number of bits in the code, print the sequence of gray code. A gray code sequence must begin with 0.

For example, given  $n = 2$ , return  $[0, 1, 3, 2]$ . Its gray code sequence is:

00	-	0
01	-	1
11	-	3
10	-	2

### Note:

For a given  $n$ , a gray code sequence is not uniquely defined.

For example,  $[0, 2, 3, 1]$  is also a valid gray code sequence according to the above definition.

For now, the judge is able to judge based on one instance of gray code sequence. Sorry about that.

» [Solve this problem](#)

### [解题思路]

看到这个题时, 首先做了一个模拟, 当 $n=3$ 时, gray code应该是

```
000
001
011
010
110
100
101
111
```

看了半天, 也没看出来什么规律。后来上网一查

GrayCode([http://en.wikipedia.org/wiki/Gray\\_code](http://en.wikipedia.org/wiki/Gray_code))才发现原来推导的gray code顺序错了。第六个应该是111。

$n=3$ 时, 正确的GrayCode应该是

```
000
001
011
010
110
111
```

### 最近阅读

- [Swift: A Storage Architecture for Large Objects](#)
- [Swift: Using Distributed Disk Striping to Provide High I/O Data Rates](#)
- [Software-defined Storage - IBM](#)

### Google Plus

[Lei Zhang](#)

### About Me



zhang lei

小磊哥

[View my complete profile](#)

[Follow @codingtmd](#)

### Visit Map

### Labels

- [Algorithm](#) (138)
- [BFS](#) (1)
- [binary tree](#) (7)
- [Bit Operation](#) (1)
- [Brute-force](#) (2)
- [design](#) (1)
- [DFS](#) (3)
- [DP](#) (16)
- [facebook](#) (1)
- [google](#) (1)

111 //如果按照题意的话，只是要求有一位不同，这里也可以是100  
101  
100

这样的话，规律就出来了， $n=k$ 时的Gray Code，相当于 $n=k-1$ 时的Gray Code的逆序 加上  $1 \ll k$ 。

[Code]

```
1: vector<int> grayCode(int n) {
2:     // Start typing your C/C++ solution below
3:     // DO NOT write int main() function
4:     vector<int> result;
5:     result.push_back(0);
6:     for(int i=0; i< n; i++)
7:     {
8:         int highestBit = 1<<i;
9:         int len = result.size();
10:        for(int i = len-1; i>=0; i--)
11:        {
12:            result.push_back(highestBit + result[i]);
13:        }
14:    }
15:    return result;
16: }
```

[总结]

题意不清楚，如果每次只是与上一个数有一个位不同的话，其实有很多种组合出来。如果不是查了Gray Code的定义，根本看不出来什么规律。

而且，Gray Code这种东西，必然有数学解，否则在早期的工程界是没法应用的。想了一下，其实也可以这么做，第 $i$ 个数可以由如下公式产生： $(i > 1) \wedge i$ ，所以代码也可以是：

```
1: vector<int> grayCode(int n)
2: {
3:     vector<int> ret;
4:     int size = 1 << n;
5:     for(int i = 0; i < size; ++i)
6:         ret.push_back((i >> 1) ^ i);
7:     return ret;
8: }
```

不过这种数学解就失去了interview的意思了。

Posted by zhang lei at 2:37 PM



[Post a Comment](#)

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

- [hash](#) (1)
- [LeetCode](#) (121)
- [Math](#) (1)
- [recursion](#) (4)
- [review](#) (22)
- [sort](#) (1)
- [SRM](#) (6)
- [TopCoder](#) (7)
- [two-pointer](#) (5)
- [Ubuntu](#) (1)
- [Yahoo](#) (1)
- [二分搜索](#) (7)
- [二叉树](#) (15)
- [动态规划](#) (20)
- [双指针](#) (16)
- [图](#) (1)
- [字符串处理](#) (8)
- [实现题](#) (7)
- [数组](#) (8)
- [模拟](#) (23)
- [解题报告](#) (22)
- [贪心](#) (1)
- [递归](#) (32)
- [递归+剪枝](#) (2)
- [链表](#) (10)

## Blog Archive

► [2013](#) (102)

▼ [2012](#) (50)

▼ [December](#) (50)

[\[LeetCode\] Roman To Integer](#)  
[解题报告](#)

[\[LeetCode\] Reverse Nodes in k-Group](#) [解题报告](#)

[\[LeetCode\] Reverse Linked List II](#) [解题报告](#)

[\[LeetCode\] Reverse Integer](#) [解题报告](#)

[\[LeetCode\] Restore IP Addresses](#) [解题报告](#)

[\[LeetCode\] Remove Nth Node From End of List](#) [解题报告](#)

[\[LeetCode\] Remove Element](#) [解题报告](#)

[\[LeetCode\] Remove Duplicates from Sorted List](#)  
[解题报告...](#)

[\[LeetCode\] Remove Duplicates from Sorted Array II](#) ...

[\[LeetCode\] Remove Duplicates from Sorted Array](#) [解题报告...](#)

[\[LeetCode\] Pow\(x, n\)](#) [解题报告](#)

[\[LeetCode\] Recover Binary Search Tree](#) [解题报告](#)

[\[LeetCode\] Populating Next Right Pointers in Each ...](#)

[\[LeetCode\] Populating Next](#)