1. 题目
2. 题干

给定一个正整数 n（1 ≤ n ≤ 30），输出外观数列的第 n 项。

注意：整数序列中的每一项将表示为一个字符串。

「外观数列」是一个整数序列，从数字 1 开始，序列中的每一项都是对前一项的描述。前五项如下：

1. 1

2. 11

3. 21

4. 1211

5. 111221

第一项是数字 1

描述前一项，这个数是 1 即 “一个 1 ”，记作 11

描述前一项，这个数是 11 即 “两个 1 ” ，记作 21

描述前一项，这个数是 21 即 “一个 2 一个 1 ” ，记作 1211

描述前一项，这个数是 1211 即 “一个 1 一个 2 两个 1 ” ，记作 111221

1. 示例

示例 1:

输入: 1

输出: "1"

解释：这是一个基本样例。

示例 2:

输入: 4

输出: "1211"

解释：当 n = 3 时，序列是 "21"，其中我们有 "2" 和 "1" 两组，"2" 可以读作 "12"，也就是出现频次 = 1 而 值 = 2；类似 "1" 可以读作 "11"。所以答案是 "12" 和 "11" 组合在一起，也就是 "1211"。

1. 题解
2. 思路

不要被题目给出的多个数列而吓到，从现象看本质，这就是一个求字符串中每个数字重复次数的问题，不过要通过重复调用而得到最终的结果。

首先，考虑n=1和n>1的区别，n=1时直接返回1，n>1时需要继续递归。在计算数字个数的函数中，主要思路就是从第一个字符开始，判断后面的字符是否等于前面的字符，如果等于，则计数count++，并且指针后移；如果不等于，则将count和上一个数存入输出字符串中。注意，最后循环出来后，最后一个字符也需要进行计数！！！

1. 代码实现

C++；

class Solution {

public:

    string countAndSay(int n) {

        string s = "1";

        if( n == 1){

            return s;

        }else{

            for(int i = 2; i <= n; ++i){

                s = toConvert(s);

            }

            return s;

        }

    }

    //计算字符个数

    string toConvert(string s){

        int len = s.size();

        char c = s[0];

        int i = 0;

        string out = "";//输出字符串

        int count = 0;  //相同字符的计数

        while( i < len){

            if(s[i] == c){

                count++;

                i++;

            }

            else{

                out = out + to\_string(count) + c;

                c = s[i];

                count = 1;

                i++;

            }

        }

        //最后一个字符也要计数

        out =  out + to\_string(count) + s[i-1];

        return out;

    }

};