Problem 38: Count and Say

Problem Information

Difficulty: Medium
Acceptance Rate: 61.73%
Paid Only: No
Tags: String

Problem Description

The **count-and-say** sequence is a sequence of digit strings defined by the recursive formula:

* `countAndSay(1) = "1"` * `countAndSay(n)` is the run-length encoding of `countAndSay(n - 1)`.

[Run-length encoding](http://en.wikipedia.org/wiki/Run-length_encoding) (RLE) is a string compression method that works by replacing consecutive identical characters (repeated 2 or more times) with the concatenation of the character and the number marking the count of the characters (length of the run). For example, to compress the string `"3322251"` we replace `"33"` with `"23"`, replace `"222"` with `"32"`, replace `"5"` with `"15"` and replace `"1"` with `"11"`. Thus the compressed string becomes `"23321511"`.

Given a positive integer `n`, return _the_`nth` _element of the**count-and- say** sequence_.

Example 1:

Input: n = 4

Output: "1211"

Explanation:

countAndSay(1) = "1" countAndSay(2) = RLE of "1" = "11" countAndSay(3) = RLE of "11" = "21" countAndSay(4) = RLE of "21" = "1211"

Example 2:

Input: n = 1

Output: "1"

Explanation:

This is the base case.

Constraints:

* `1 <= n <= 30`

Follow up: Could you solve it iteratively?

Code Snippets

C++:

```
class Solution {
public:
string countAndSay(int n) {

}
};
```

Java:

```
class Solution {
public String countAndSay(int n) {

}
}
```

Python3:

```
class Solution:
def countAndSay(self, n: int) -> str:
```