118 Pascal's Triangle

2018年4月1日 14:28

Question:

Given *numRows*, generate the first *numRows* of Pascal's triangle.
For example, given *numRows* = 5,
Return
[
[1],
[1,1],
[1,2,1],
[1,3,3,1],
[1,4,6,4,1]
]

来自 <https://leetcode.com/problems/pascals-triangle/description/>

给定 *numRows*，生成帕斯卡三角形的前 *numRows* 行。

Solution for Python3:

```python
class Solution1:
    def generate(self, numRows):
        """
        :type numRows: int
        :rtype: List[List[int]]
        """
        L = []
        for i in range(numRows):
            L.append([1])
            for j in range(1, i):
                L[i].append(L[i - 1][j - 1] + L[i - 1][j])
            if i > 0:
                L[i].append(1)
        return L

class Solution2:
    def generate(self, numRows):
        """
        :type numRows: int
        :rtype: List[List[int]]
        """
        triangle = []
        for row_num in range(numRows):
            row = [None for _ in range(row_num + 1)]
            row[0], row[-1] = 1, 1
            for j in range(1, len(row) - 1):
                row[j] = triangle[row_num - 1][j - 1] + triangle[row_num - 1][j]
            triangle.append(row)
        return triangle
```

Solution for C++:

```cpp
class Solution {
public:
    vector<vector<int>> generate(int numRows) {
```

```
        vector<vector<int> > v(numRows);
        for (int i = 0; i < numRows; i++) {
            v[i].resize(i + 1);     //每行数量固定
            v[i][0] = v[i][i] = 1;  //每行首尾固定
            for(int j = 1; j < i; j++) {//每行中间部分
                v[i][j] = v[i - 1][j - 1] + v[i - 1][j];
            }
        }
        return v;
    }
};
```