1. 题目

（一）概述

将一个给定字符串根据给定的行数，以从上往下、从左到右进行 Z 字形排列。

比如输入字符串为 "LEETCODEISHIRING" 行数为 3 时，排列如下：

L C I R

E T O E S I I G

E D H N

之后，你的输出需要从左往右逐行读取，产生出一个新的字符串，比如："LCIRETOESIIGEDHN"。

请你实现这个将字符串进行指定行数变换的函数：

string convert(string s, int numRows);

（二）示例

示例 1:

输入: s = "LEETCODEISHIRING", numRows = 3

输出: "LCIRETOESIIGEDHN"

示例 2:

输入: s = "LEETCODEISHIRING", numRows = 4

输出: "LDREOEIIECIHNTSG"

解释:

L D R

E O E I I

E C I H N

T S G

二．题解

（一）思路

1. 通过题目得知，将字符串按照Z字形排列后，从左到右，输出每一行的字符。因此，想到每一行的字符串都是分开存储的。将每一行的字符串存储在一个字符串数组rows中。

s是字符串，numRows是行数，如果行数为1，则直接返回原字符串s。

2. 若字符串的长度小于numRows，说明字符串连第一列都没排完，则rows数组只需存储s.size()个字符串即可；若字符串的长度大于或等于numRows，说明第一列排完了，可以继续排列，rows数组需存储numRows个字符串。

3. 模拟排列的场景，可知连续字符的行数都是递增或者递减的，当向下排列时，则行数递增；当向上排列时，则行数递减。我们可以设置一个flag，表示排列的方向。向下时为+1，行数+1；向上时为-1，行数-1。

4. 注意，有边界的问题。当字符到了最后一行时，方向需要变化，flag需要改变。

5. 我们要做的是以行数为索引，为每一行的字符串添加当前的字符。最后再遍历一次rows，将所有的字符串都连接起来，返回字符串。

（二）代码实现

C++

#include<string>

#include<vector>

#include<algorithm>

using namespace std;

class Solution {

public:

string convert(string s, int numRows) {

if (numRows == 1)

return s;

//创建字符串数组

vector<string> rows(min(numRows, int(s.size())));

int curRow = 0;

int flag = 1;

for (char c : s) {

rows[curRow] += c;

if (curRow == 0 || curRow == numRows - 1)

flag = -flag;

curRow += flag;

}

string ret;

for (string row : rows)

ret += row;

return ret;

}

};

Java：

import java.util.ArrayList;

import java.util.List;

public class Solution {

    public String convert(String s, int numRows){

        if(numRows == 1)

            return s;

        List<StringBuilder> rows = new ArrayList<>();

        for(int i = 0; i < Math.min(numRows, s.length()); ++i)

            rows.add(new StringBuilder());

        int curRow = 0;

        int flag = -1;

        for(char c : s.toCharArray()){

            rows.get(curRow).append(c);

            if(curRow == 0 || curRow == numRows-1)

                flag = -flag;

            curRow += flag;

        }

        StringBuilder ret = new StringBuilder();

        for(StringBuilder row : rows)

            ret.append(row);

        return ret.toString();

    }

}