

## 6. ZigZag Conversion

June 11, 2018 | 189.9K views

PreviousNext

★★★★★  
Average Rating: 3.90 (105 votes)

The string "PAYPALISHIRING" is written in a zigzag pattern on a given number of rows like this: (you may want to display this pattern in a fixed font for better legibility)

```
P   A   H   N
A P L S I I G
Y   I   R
```

And then read line by line: "PAHNAPLSIIGYIR"

Write the code that will take a string and make this conversion given a number of rows:

```
string convert(string s, int numRows);
```

Example 1:

Input: s = "PAYPALISHIRING", numRows = 3

Output: "PAHNAPLSIIGYIR"

Example 2:

Input: s = "PAYPALISHIRING", numRows = 4

Output: "PINALSIGYAHRPI"

Explanation:

```
P       I       N
A    L S   I G
Y A  H R
P    I
```

## Solution

### Approach 1: Sort by Row

#### Intuition

By iterating through the string from left to right, we can easily determine which row in the Zig-Zag pattern that a character belongs to.

#### Algorithm

We can use  $\min(\text{numRows}, \text{len}(s))$  lists to represent the non-empty rows of the Zig-Zag Pattern.

Iterate through  $s$  from left to right, appending each character to the appropriate row. The appropriate row can be tracked using two variables: the current row and the current direction.

The current direction changes only when we moved up to the topmost row or moved down to the bottommost row.

```
C++JavaCopy
1 class Solution {
2 public:
3     string convert(string s, int numRows) {
4
5         if (numRows == 1) return s;
6
7         vector<string> rows(min(numRows, int(s.size())));
8         int curRow = 0;
9         bool goingDown = false;
10
11         for (char c : s) {
12             rows[curRow] += c;
13             if (curRow == 0 || curRow == numRows - 1) goingDown = !goingDown;
14             curRow += goingDown ? 1 : -1;
15         }
16
17         string ret;
18         for (string row : rows) ret += row;
19         return ret;
20     }
21 };
```

#### Complexity Analysis

- Time Complexity:  $O(n)$ , where  $n == \text{len}(s)$
- Space Complexity:  $O(n)$

### Approach 2: Visit by Row

#### Intuition

Visit the characters in the same order as reading the Zig-Zag pattern line by line.

#### Algorithm

Visit all characters in row 0 first, then row 1, then row 2, and so on...

For all whole numbers  $k$ ,

- Characters in row 0 are located at indexes  $k(2 \cdot \text{numRows} - 2)$
- Characters in row  $\text{numRows} - 1$  are located at indexes  $k(2 \cdot \text{numRows} - 2) + \text{numRows} - 1$
- Characters in inner row  $i$  are located at indexes  $k(2 \cdot \text{numRows} - 2) + i$  and  $(k + 1)(2 \cdot \text{numRows} - 2) - i$ .

```
C++JavaCopy
1 class Solution {
2 public:
3     string convert(string s, int numRows) {
4
5         if (numRows == 1) return s;
6
7         string ret;
8         int n = s.size();
9         int cycleLen = 2 * numRows - 2;
10
11         for (int i = 0; i < numRows; i++) {
12             for (int j = 0; j + i < n; j += cycleLen) {
13                 ret += s[j + i];
14                 if (i != 0 && i != numRows - 1 && j + cycleLen - i < n)
15                     ret += s[j + cycleLen - i];
16             }
17         }
18         return ret;
19     }
20 };
```

#### Complexity Analysis

- Time Complexity:  $O(n)$ , where  $n == \text{len}(s)$ . Each index is visited once.
- Space Complexity:  $O(n)$ . For the cpp implementation,  $O(1)$  if return string is not considered extra space.

Rate this article: ★★★★★

PreviousNext

Comments: 152Sort By

Type comment here... (Markdown is supported)

PreviewPost

Ronnie268 ★56 December 24, 2018 2:53 AM

Python3 (beats 95%):

```
class Solution:
    def convert(self, s, numRows):
        """
```

Read More

55 ^ v | Share | Reply

SHOW 2 REPLIES

lwxiaolei ★58 August 13, 2018 12:54 PM

python solution:

```
class Solution(object):
    def convert(self, s, numRows):
        if numRows == 1:
```

Read More

44 ^ v | Share | Reply

SHOW 3 REPLIES

cptdav ★34 August 16, 2018 10:58 PM

a python solution using O(1) space and O(n) time:

```
class Solution(object):
    def convert(self, s, numRows):
        if s == None:
```

Read More

32 ^ v | Share | Reply

SHOW 4 REPLIES

NaT3z ★24 October 23, 2018 7:39 PM

Simple Python3 solution using dict/hashtable. I like it because it's intuitive and the "step" between rows can be visualised. 76ms runtime (99.62th percentile).

try except block without specific error (KeyError to avoid string length < numRows) might be bad form but specifying strangely increased runtime considerably.

Read More

19 ^ v | Share | Reply

SHOW 3 REPLIES

anthonyw ★10 December 27, 2018 7:00 PM

JavaScript:

```
var convert = function(s, numRows) {
    var length = s.length;
    if (numRows == 1) return s;
```

Read More

10 ^ v | Share | Reply

h11129 ★16 October 4, 2018 9:41 AM

I think the solution is actually not zigzag as what's told in the problem. It work because we only need to print out the zigzag line by line. It won't work if the question is to print out the zigzag. It lose the shape.

14 ^ v | Share | Reply

SHOW 3 REPLIES

zkf85 ★39 March 28, 2019 7:19 AM

Python Solution:

```
class Solution:
    def convert(self, s: str, numRows: int) -> str:
        if numRows == 1:
```

Read More

12 ^ v | Share | Reply

mrooo ★13 September 24, 2018 6:06 PM

```
class Solution {
    public String convert(String s, int numRows) {
        //如果是有一行，可以直接返回
        if (numRows == 1) {
```

Read More

12 ^ v | Share | Reply

wguo32 ★55 September 17, 2018 12:59 AM

my clumsy answer:

```
class Solution {
    public String convert(String s, int numRows) {
        //null check
```

Read More

9 ^ v | Share | Reply

SHOW 3 REPLIES

user7579F ★5 January 12, 2019 1:59 AM

Python 3 solution using a generator:

```
class Solution:
    def convert(self, s, numRows):
        """
```

Read More

5 ^ v | Share | Reply

< 1 2 3 4 5 6 ... 15 16 >