Zigzag Conversion

For solving the Zigzag Conversion problem, the goal is to simulate writing characters in a zigzag pattern and then reading them row by row.

Core Idea: Simulate the zigzag by filling characters into rows going down and then up diagonally repeatedly.

Use:
- A vector<string> to store characters for each row.
- A row index and a direction flag to determine movement between rows.

Algorithm:
1. If numRows == 1 → return s directly (zigzag isn't needed)
2. Initialize a vector of empty strings: vector<string> rows(numRows)
3. Use row = 0, and direction = -1
4. Loop through characters in s:
   - Append to rows[row]
   - If at top or bottom row, reverse direction
   - Move row += direction
5. After loop, concatenate all strings in rows

Time Complexity:
- O(n) where n = s.length() – each character is processed once
- O(n) space for storing rows.

Example: "PAYPALISHIRING", numsRows = 4

Row 0: P I N
Row 1: A L S I G
Row 2: Y A H R
Row 3: P I

Concatenate: "PINALSIGYAHRPI"