Solution@05-12-23

2022. Convert 1D Array Into 2D Array for java

```
// Online Java Compiler
// Use this editor to write, compile and run your Java code online

import java.util.Scanner;

public class ConvertArray {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Get the number of rows and columns for the 2D array
        System.out.print("Enter the number of rows: ");
        int rows = scanner.nextInt();
        System.out.print("Enter the number of columns: ");
        int cols = scanner.nextInt();

        // Get the elements of the 1D array
        System.out.println("Enter the elements of the 1D array separated by spaces:");
        int[] oneDArray = new int[rows * cols];
        for (int i = 0; i < oneDArray.length; i++) {
            oneDArray[i] = scanner.nextInt();
        }

        // Convert 1D array to 2D array
        int[][] twoDArray = convertTo2DArray(oneDArray, rows, cols);

        // Print the 2D array
```

```
        System.out.println("Converted 2D array:");
        print2DArray(twoDArray);
    }

    // Function to convert 1D array to 2D array
    private static int[][] convertTo2DArray(int[] oneDArray, int rows, int cols) {
        int[][] twoDArray = new int[rows][cols];
        int index = 0;

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                twoDArray[i][j] = oneDArray[index++];
            }
        }

        return twoDArray;
    }

    // Function to print 2D array
    private static void print2DArray(int[][] twoDArray) {
        for (int i = 0; i < twoDArray.length; i++) {
            for (int j = 0; j < twoDArray[i].length; j++) {
                System.out.print(twoDArray[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Output

```
PS C:\Users\Ajeet\Desktop\java> javac ConvertArray.java
```

```
PS C:\Users\Ajeet\Desktop\java> java ConvertArray
Enter the number of rows: 3
Enter the number of columns: 4
Enter the elements of the 1D array separated by spaces:
1 2 3 4 5 6
1 2 3 4 5 6
Converted 2D array:
1 2 3 4
5 6 1 2
3 4 5 6
```

1541. Minimum Insertions to Balance a Parentheses String

```
import java.util.Scanner;
import java.util.Stack;

public class MinimumInsertions {

    public static int minInsertions(String s) {
        int insertions = 0;
        Stack<Character> stack = new Stack<>();

        for (char c : s.toCharArray()) {
            if (c == '(') {
                stack.push(c);
            } else {
                if (stack.isEmpty()) {
                    insertions++;
                } else {
                    stack.pop();
                }

                if (stack.isEmpty()
```

```
                && (c == ')' && (s.indexOf(c) + 1 == s.length() || s.charAt(s.indexOf(c) + 1) != ')'))) {
                insertions++;
            }
        }
    }

    insertions += stack.size() * 2;

    return insertions;
  }

  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter the parentheses string: ");
    String input = scanner.nextLine();

    int result = minInsertions(input);
    System.out.println("Minimum insertions needed: " + result);

    scanner.close();
  }
}
```

output

```
PS C:\Users\Ajeet\Desktop\java> javac MinimumInsertions.java
PS C:\Users\Ajeet\Desktop\java> java MinimumInsertions
Enter the parentheses string: ((((((((((
Minimum insertions needed: 2
```