CodeIT Suisse 2022

Welcome to CodeIT Suisse 2022

Social Distancing - 2022

Challenge description

The outbreak of COVID-19 has affected our daily lives for over 2 years. Wherever we go, social distancing is implemented to reduce the transmission of the virus.

In this challenge, you are provided with a venue and the number of people who will occupy it. Some of the visitors unknowingly have COVID, and have already taken their seats.

Your mission, should you wish to accept it, is to come up with the seating plans so that the rest of the visitors are seated while respecting social distancing.

In our venue, a socially distanced seat should not have another person next to them (in any direction).

Each visitor is denoted by `V`.

Endpoint

We will need you to expose the following endpoint for us to evaluate your challenge:

```
POST /social-distancing
```

Input

You will receive a `JSON` payload that contains an array of Strings. Each entry in the array denotes a "venue" with its social distancing constraints.

Simple example input

```
[
  "3, 3, 2",
  "1, 1, 10"
]
```

Complex example input

```
[
  "4, 3, 3, 0, 3, 2, 1",
  "4, 3, 3, 0, 0, 0, 1"
]
```

Inputs explained

```
"width, height, numberOfVisitors, occupiedSeats..."
```

width (`number`):

- the width of venue (how many columns it has)
- ***must be*** > 0

height (`number`):

- the height of venue (how many rows it has)
- ***must be*** > 0

numberOfVisitors (`number`):

- the total number of people attending the venue
- ***must be*** > 0

***(Optional)* occupiedSeats (`row, column, row, column, ...` or ``):**

- **optional arguments**
- every additional pair of arguments will correspond to a row and column representing occupied seats
- row is a `number`
- column is a `number`
- See examples below for more details

Output

We expect to receive a `JSON` formatted array as output.

The array of outputs should be returned in the same order as the inputs.

E.g.:

```
// Input JSON:
[
  "3, 2, 2",
  "1, 1, 10",
  "4, 3, 3, 0, 0, 0, 1"
]


// Output JSON:
[4, "No Solution", 6]
```

Rules / Constraints

Timeouts

- If your solution times out, you will likely score `0`

Brute forcing

- Very unlikely that you'll be able to complete this challenge with a brute force approach

Space constraints

- Be wary of running out of memory when solving this challenge

Scoring

- You will be given 10 problems in the `JSON` array
- Each problem will be scored as correct or incorrect
- There are no partial scores for each problem
- E.g., maximum score is 10 out of 10, but could score N out of 10.

Examples

Example 1

```
Input:
"3, 2, 2"

Output:
4

Explanation: [
  [
    ["V", ".", "V"],
    [".", ".", "."]
  ],[
    ["V", ".", "."],
    [".", ".", "V"],
  ],[
    [".", ".", "V"],
    ["V", ".", "."],
  ],[
    [".", ".", "."],
    ["V", ".", "V"]
  ]
]
```

Example 2

```
Input:
"1, 1, 10"
```

```

Output:
No Solution
```

Example 3

```
Input:
"4, 3, 3, 0, 3, 2, 1"

Output:
3

Explanation: After the first 3 arguments, remaining are the occupied seat coordinat
[0,3] and [2,1]

[
  [
    [".", ".", ".", "V"],
    [".", ".", ".", "."],
    [".", "V", ".", "V"],
  ],[
    ["V", ".", ".", "V"],
    [".", ".", ".", "."],
    [".", "V", ".", "."],
  ],[
    [".", "V", ".", "V"],
    [".", ".", ".", "."],
    [".", "V", ".", "."],
  ]
]
```

Example 4

Note: When two `occupiedSeats` are next to each other, you can still get valid solutions, as other attendees can be seated at a safe distance from them.

```
Input:
"4, 3, 3, 0, 0, 0, 1"

Output:
6

Explanation: [
  [
    ["V", "V", ".", "V"],
    [".", ".", ".", "."],
    [".", ".", ".", "."]
  ], [
    ["V", "V", ".", "."],
    [".", ".", ".", "V"],
    [".", ".", ".", "."]
  ], [
    ["V", "V", ".", "."],
    [".", ".", ".", "."],
    ["V", ".", ".", "."]
  ], [
    ["V", "V", ".", "."],
    [".", ".", ".", "."],
    [".", "V", ".", "."]
  ], [
    ["V", "V", ".", "."],
    [".", ".", ".", "."],
    [".", ".", "V", "."]
  ], [
    ["V", "V", ".", "."],
    [".", ".", ".", "."],
    [".", ".", ".", "V"]
  ]
]
```