**Treasure Hunt**

There is a treasure hunt game organised in a land with a grid of NxM plots

In this game, an N by M grid of plots was covered by soil, and some treasures were hidden under the soil.

During the initial setup for the game, the game designer would randomly place some treasures on the grid. Now every plot is in one of three states:

1. It has a treasure.
2. It is adjacent to a treasure. Adjacent means - the plot is either horizontally, vertically, or diagonally neighbour to a treasure. Its value will be the count of how many treasures it is adjacent to.
3. It is not adjacent to any treasure. Its value will be zero

Your job will be to show all the plots with correct values.

You have to ask for grid size first and a ‘create grid button’.

Then show the grid of that size to get the treasures, i.e. there should be N x M plots with empty values.

User will click on a plot to mark it with the treasure. The value in the plot should change to ‘x’ Clicking again should remove the treasure from that plot.

Once user marks the plots with treasures, there should be a submit button to show another grid with correct values in **all** the plots.

Create a reset button to start the exercise again

Example:

Enter the number of rows in grid : [3]

Enter the number of columns in grid: [3]

[Create grid]

Enter the treasures :

|  |  |  |
| --- | --- | --- |
| x | x |  |
|  |  |  |
|  |  |  |

[Submit]

|  |  |  |
| --- | --- | --- |
| x | x | 1 |
| 2 | 2 | 1 |
| 0 | 0 | 0 |

[Reset]

Few points to take care:

1. Your algorithm to compute the values of plots should be clear and optimized. This is one of the key criterias for judging. It should be written in **Javascript.**
2. Don’t focus on design, don’t use any CSS themes, keep it simple. This problem is to understand your logical thinking.
3. Reusable code is the key to bug free application.

Submit the zip folder including a README file. README should have clear instructions of how to run it in browser.