

# Input Taking

1. I took the given terrain and transformed it into a 2d array.

# Printing the terrain

1. Printed the terrain with the right amount of spaces between characters .

# Output giving

1. I assumed the outputs I took was right and the if the output given was not right , with try and catch methods , I send the same methods for taking the desired input .
2. When I get the right input , I transform it into numbers and make the necessary modifications to my terrain
3. I print the terrain as wanted in the game

# Solving the game

1. Firstly , I assumed as the water is not fluid and created a matrix with the water level equal to the biggest height possible (biggest number in the terrain ) , we can think as creating a rectangular prism. Let's call this flooded terrain.
2. Then , I derived a matrix to find the added water for each coordinate of the terrain , in order to find that , for each coordinate , I subtracted the terrain from the flooded terrain . We can call this new terrain added water terrain
3. For purposes later discussed , I created a method which erases the water (erases just 1 unit) that is connected to the given coordinate , and if it calls the same method when it erases a water unit , so what this method does is erases 1 unit from a water group that is connected, from the flooded terrain and the added water terrain. Let's call this method eliminating water method.
  - For understanding if the water is erasable or not , I used added water terrain to check if there is water . For understanding the height I checked the flooded terrain .
    - \* if there is water and the height is more I erased enough water to balance the heights .
4. At last , I used the eliminating water method to eliminate all waters added to the sides , because at the sides, there can't be any water.

# Printing the final terrain with water lakes and printing the result

1. I checked the added water terrain if there were any water , if so , I added the water units of a lake to a sum variable .
2. Then when i summed up all the units of a lake with a recursive algorithm , i added the square root of this sum variable to a general sum.
3. While using this recursive algorithm i created a new terrain with alphabetical characters at the coordinates where the lakes found .
4. I printed out the new terrain and below that i printed out the general sum