# Problem 2 – Parachute

You find yourself in a training for being the **best parachute jumper** in the world. At the beginning of the training you need to understand how **gravity** and **wind** work. You are given all the data from previous jumps of your colleagues. Your task is to determine how the **jumper** will **finish** his jump and **where** he will **land** exactly, based on the gravity and wind parameters.

You are given an **array of strings**. The **jumper** can be **anywhere** in the array and is denoted by the **"o" symbol**. You need to determine the **movement** of the jumper in **iterations**. On each iteration the jumper moves **one line down**, pulled by **gravity**. Additionally, the jumper moves **sideways** by the **wind** on the **current** line.

* The **">" symbol** means the wind is moving the jumper **one position** to the **right**.
* The **"<" symbol** means the wind is moving the jumper **one position** to the **left**.

The **total direction** of the wind on a single line may **stack** (e.g. "**>>>**" moves the jumper 3 positions to the right; "**><<**" moves the jumper 1 position to the left).

See examples to better understand the motion of the jumper.

The jumper can move only through **air** (the **"-" symbol**). When the jumper hits the **ground**, **water** or a **cliff**, the jump is **finished** and you need to **print the outcome** of the jump.

### Input

The input will be passed to the first JavaScript function found in your code as **array of strings**, each containing a **symbol**. The **symbols** are **not separated** by anything. The input data will always be valid and in the format described.

### Output

The output consists of two lines. The first line holds a string: **"[landing place] ". There are 3 possible outcome messages:**

* If you have landed on the **ground** ("\_" symbol), you are well and alive: **"Landed on the ground like a boss!"**
* If you have landed in the **water** ("~" symbol), you have drowned: "**Drowned in the water like a cat!"**
* If you have landed on **a cliff** ("/", "\" or "|" symbol), you have died: "**Got smacked on the rock like a dog!**

The second line holds the **position** (the **row** and **col**)of your landing.

### Constraints

* The **row** and **col** of the matrix will be in the range **[0…10]**.
* The jumper will never fly off the map.
* Time limit: 0.3 sec. Memory limit: 16 MB.

**function** *solve*(input) {  
  
 **let** startCol = 0;  
 **let** startRow = 0;  
 **let** countRight = 0;  
 **let** countLeft = 0;  
 **let** matrix = input;  
  
 **for** (**let** i = 0; i < matrix.length; i++) {  
  
 **let** column = matrix[i].indexOf(**'o'**);  
 **if** (column > 0) {  
 startCol = matrix[i][column];  
 startRow = i;  
 startCol = column;  
 **break**;  
 }  
 }  
  
 **for** (**let** i = startRow+1; i < matrix.length; i++) {  
 countLeft = 0;  
 countRight = 0;  
  
 **for** (**let** j = 0; j < matrix[i].length; j++) {  
  
 **if**(matrix[i][j] == **'>'**){  
 countRight++;  
 }  
 **if**(matrix[i][j] == **'<'**){  
 countLeft++;  
 }  
 }  
  
 **if**(countRight > countLeft){  
 startRow += 1;  
 startCol += countRight-countLeft;  
 }  
 **else if**(countRight < countLeft){  
 startRow += 1;  
 startCol = startCol - (countLeft - countRight);  
 }  
 **else**{  
 startRow += 1;  
 }  
 }  
 **if**(matrix[startRow][startCol] == **'\_'**){  
 **console**.log(**"Landed on the ground like a boss!"**);  
 **console**.log(**''** + startRow + **' '** + startCol)  
 }  
 **else if**(matrix[startRow][startCol] == **'~'**){  
 **console**.log(**"Drowned in the water like a cat!"**);  
 **console**.log(**''** + startRow + **' '**+ startCol)  
 }  
 **else if**((matrix[startRow][startCol] == **'/'**) ||(matrix[startRow][startCol] == **'\\'**) ||  
 (matrix[startRow][startCol] == **'|'**)){  
 **console**.log(**"Got smacked on the rock like a dog!"**);  
 **console**.log(**''** + startRow + **' '** + startCol)  
 }  
 **console**.log(**''** + startRow + **' '** + startCol)  
  
}  
  
*solve*([  
 **'o----------/-------------'**,  
 **'-------->>/-->>-->><<----'**,  
 **'----------\--->>>>-------'**,  
 **'-----------\\_\_\_\_\_\_\_\_\_>---'**,  
 **'---------------------\>--'**,  
 **'>>>>>>----------------\-<'**,  
 **'------>>>>------------/--'**,  
 **'---------------------/---'**,  
 **'--------------------/----'**,  
 **'-------------------/-----'**]);

### Examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| --o----------------------  >------------------------  >------------------------  >-----------------/\-----  -----------------/--\----  >---------/\----/----\---  ---------/--\--/------\--  <-------/----\/--------\-  \------/----------------\  -\\_\_\_\_/------------------ | Landed on the ground like a boss!  9 5 |  | -------------o-<<--------  -------->>>>>------------  ---------------->-<---<--  ------<<<<<-------/\--<--  --------------<--/-<\----  >>--------/\----/<-<-\---  ---------/<-\--/------\--  <-------/----\/--------\-  \------/--------------<-\  -\\_\_\_~/------<----------- | Drowned in the water like a cat!  9 5 |