We have a collection of rocks, each rock has a positive integer weight.

Each turn, we choose **any two rocks** and smash them together.  Suppose the stones have weights x and y with x <= y.  The result of this smash is:

* If x == y, both stones are totally destroyed;
* If x != y, the stone of weight x is totally destroyed, and the stone of weight y has new weight y-x.

At the end, there is at most 1 stone left.  Return the **smallest possible** weight of this stone (the weight is 0 if there are no stones left.)

**Example 1:**

**Input:** [2,7,4,1,8,1]

**Output:** 1

**Explanation:**

We can combine 2 and 4 to get 2 so the array converts to [2,7,1,8,1] then,

we can combine 7 and 8 to get 1 so the array converts to [2,1,1,1] then,

we can combine 2 and 1 to get 1 so the array converts to [1,1,1] then,

we can combine 1 and 1 to get 0 so the array converts to [1] then that's the optimal value.

**Note:**

1. 1 <= stones.length <= 30
2. 1 <= stones[i] <= 100