https://course.acciojob.com/idle?question=d2ecb8ce-212e-404c-978c-ba98835e602f

- EASY
- Max Score: 30 Points

Last Stone Weight

We have a collection of 'N' stones, each stone has a positive integer weight. On each turn, we choose the two heaviest stones and smash them together. Suppose the stones have weights 'x' and 'y' with 'x' <= 'y'. The result of this smash will be:

1. **If 'x' == 'y', both stones are totally destroyed;**
2. **If 'x' != 'y', the stone of weight 'x' is totally destroyed, and the stone of weight 'y' has a new weight equal to 'y - x'.**

In the end, there is at most 1 stone left. Print the weight of this stone or 0 if there are no stones left.

Input Format

The first line of input contains the integer 'N', representing the total number of stones.

The second line of input contains 'N' single space-separated integers, representing the weights of the stones.

Output Format

The only output line prints the weight of the last stone, if it exists, 0 otherwise.

Example 1

Input

```
1
10
```

Output

```
10
```

Explanation

There is Only one stone so the weight of the last stone is 10

Example 2

Input

```
3
1 9 5
```

Output

```
3
```

Constraints

1 <= N <= 10^5 1 <= W <= 10^6

Time Limit: 1 sec

Topic Tags

- **Math**

My code

```
// n java
import java.io.*;
import java.util.*;
```

```
public class Main {
	public static void main(String args[]) {
		// your code here
		Scanner sc = new Scanner(System.in);

		int n = sc.nextInt();
		int arr[]= new int[n];
		for(int i=0;i<n;i++)
			arr[i]=sc.nextInt();

		PriorityQueue<Integer> pq    = new
PriorityQueue<Integer>( Collections.reverseOrder());

		for(int i=0;i<n;i++)
			pq.add(arr[i]);

		while(pq.size()>1)
		{
			int n1 = pq.poll();
			int n2 = pq.poll();
			if(n1!=n2)
				pq.add(Math.abs(n1-n2));
		}
		if(pq.size()==0)
			System.out.println(0);
		else
			System.out.println(pq.peek());
	}
}
```