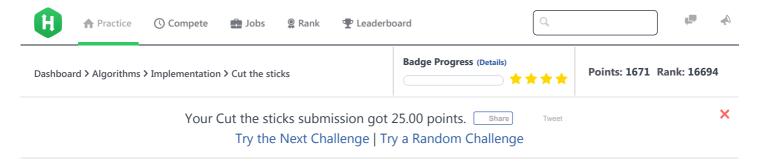
3/31/2018 HackerRank



Cut the sticks **■**



|--|

You are given a number of sticks of varying lengths. You will iteratively cut the sticks into smaller sticks, discarding the shortest pieces until there are none left. At each iteration you will determine the length of the shortest stick remaining, cut that length from each of the longer sticks and then discard all the pieces of that shortest length. When all the remaining sticks are the same length, they cannot be shortened so discard them.

Given the lengths of n sticks, print the number of sticks that are left before each iteration until there are none left.

Note: Before each iteration you must determine the current shortest stick.

Input Format

The first line contains a single integer n.

The next line contains \boldsymbol{n} space-separated integers: a_0 , a_1 ,... a_{n-1} , where $\boldsymbol{a_i}$ represents the length of the $\boldsymbol{i^{th}}$ stick in array arr.

Output Format

For each operation, print the number of sticks that are cut, on separate lines.

Constraints

- $1 \le n \le 1000$
- $1 \le a_i \le 1000$

Sample Input 0

6 5 4 4 2 2 8

Sample Output 0

6 4 2

Sample Input 1

8 1 2 3 4 3 3 2 1

Sample Output 1

8 6 4

Explanation

3/31/2018 HackerRank

```
Sample Case 0:
```

```
        sticks-length
        length-of-cut
        sticks-cut

        5 4 4 2 2 8
        2
        6

        3 2 2 _ _ 6
        2
        4

        1 _ _ _ 4
        1
        2

        - _ _ 3
        3
        1

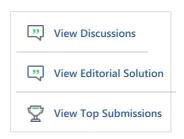
        DONE
        DONE
```

Sample Case 1

sticks-length	length-of-cut	sticks-cut
1 2 3 4 3 3 2 1	1	8
_ 1 2 3 2 2 1 _	1	6
1 2 1 1	1	4
1	1	1
	DONE	DONE



Need Help?



Rate This Challenge:



Resources

Sorting

Download problem statement

Download sample test cases

Suggest Edits

Choose a translation





```
Current Buffer (saved locally, editable) &
                                                                                      C++14
                                                                                                                        \Diamond
 1 ▼ #include <iostream>
 2 #include <map>
 3
 4 int main()
 5 ▼ {
         int n; std::cin >> n;
 6
 7
        std::map<int,int> Map;
 8
        for(int i = 0; i < n; ++i)
 9
10 ▼
        {
             int temp; std::cin >> temp;
11
12 ▼
             Map[temp]++;
13
        }
14
```

3/31/2018 HackerRank

```
15
          int sticks = n; //number of sticks that are cut
  16
          std::cout << sticks << std::endl;</pre>
  17
          if(Map.size() > 1)
  18
  19 ▼
          {
  20
               for(const auto &it: Map)
  21 ▼
  22
                  sticks -= it.second;
                  if(sticks == 0) break;
  23
                  std::cout << sticks << std::endl;</pre>
  24
  25
  26
          }
  27
          return 0;
     }
  28
  29
                                                                                                               Line: 29 Col: 1
                                                                                                                 Submit Code
                     Test against custom input
                                                                                                    Run Code
1 Upload Code as File
                                      Congrats, you solved this challenge!
                                              Challenge your friends: f y in
               ✓ Test Case #0
                                                       ✓ Test Case #1
                                                                                               ✓ Test Case #2
               ✓ Test Case #3
                                                       ✓ Test Case #4
                                                                                               ✓ Test Case #5
```

 $Contest\ Calendar |Blog|Scoring|Environment|FAQ|About\ Us|Support|Careers|Terms\ Of\ Service|Privacy\ Policy|Request\ a\ Feature$

✓ Test Case #7

✓ Test Case #8

Next Challenge

You've earned 25.00 points.

✓ Test Case #6

✓ Test Case #9