Qualification Round 2011

Contest Analysis

Questions asked 3

▬ Submissions

Bot Trust

10pt	Not attempted **10560/12572 users** correct (84%)
10pt	Not attempted **10291/10514 users** correct (98%)

Magicka

10pt	Not attempted **8886/10218 users** correct (87%)
15pt	Not attempted **7176/8738 users** correct (82%)

Candy Splitting

10pt	Not attempted **8188/9096 users** correct (90%)
15pt	Not attempted **6286/7416 users** correct (85%)

GoroSort

10pt	Not attempted **2670/4609 users** correct (58%)
20pt	Not attempted **2568/2649 users** correct (97%)

▬ Top Scores

SkidanovAlexander	100
tomconerly	100
kmod	100
watashi	100
RAD.	100
Anton.Lunyov	100
w01fe	100
jakubr	100
Weiqi	100
hos.lyric	100

Problem C. **Candy Splitting**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 10 points	Solve C-small
Large input 15 points	Solve C-large

Problem

Sean and Patrick are brothers who just got a nice bag of candy from their parents. Each piece of candy has some positive integer value, and the children want to divide the candy between them. First, Sean will split the candy into two piles, and choose one to give to Patrick. Then Patrick will try to calculate the value of each pile, where the value of a pile is the sum of the values of all pieces of candy in that pile; if he decides the piles don't have equal value, he will start crying.

Unfortunately, Patrick is very young and doesn't know how to add properly. He *almost* knows how to add numbers in binary; but when he adds two 1s together, he always forgets to carry the remainder to the next bit. For example, if he wants to sum 12 (1100 in binary) and 5 (101 in binary), he will add the two rightmost bits correctly, but in the third bit he will forget to carry the remainder to the next bit:

```
  1100
+ 0101
------
  1001
```

So after adding the last bit without the carry from the third bit, the final result is 9 (1001 in binary). Here are some other examples of Patrick's math skills:

```
5 + 4 = 1
7 + 9 = 14
50 + 10 = 56
```

Sean is very good at adding, and he wants to take as much value as he can without causing his little brother to cry. If it's possible, he will split the bag of candy into two non-empty piles such that Patrick thinks that both have the same value. Given the values of all pieces of candy in the bag, we would like to know if this is possible; and, if it's possible, determine the maximum possible value of Sean's pile.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case is described in two lines. The first line contains a single integer **N**, denoting the number of candies in the bag. The next line contains the **N** integers C_i separated by single spaces, which denote the value of each piece of candy in the bag.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1). If it is impossible for Sean to keep Patrick from crying, y should be the word "NO". Otherwise, y should be the value of the pile of candies that Sean will keep.

Limits

$1 \le \mathbf{T} \le 100$.
$1 \le \mathbf{C_i} \le 10^6$.

Small dataset

$2 \le \mathbf{N} \le 15$.

Large dataset

$2 \le \mathbf{N} \le 1000$.

Sample

Input	Output

```
2              Case #1: NO
5              Case #2: 11
1 2 3 4 5
3
3 5 6
```