Google Code Jam '15 Qualification Round Problem B - Infinite House of Pancakes

Time limit: 60.0s **Memory limit:** 1G

At the Infinite House of Pancakes, there are only finitely many pancakes, but there are infinitely many diners who would be willing to eat them! When the restaurant opens for breakfast, among the infinitely many diners, exactly D have non-empty plates; the i^{th} of these has P_i pancakes on his or her plate. Everyone else has an empty plate.

Normally, every minute, every diner with a non-empty plate will eat one pancake from his or her plate. However, some minutes may be *special*. In a special minute, the head server asks for the diners' attention, chooses a diner with a non-empty plate, and carefully lifts some number of pancakes off of that diner's plate and moves those pancakes onto one other diner's (empty or non-empty) plate. No diners eat during a special minute, because it would be rude.

You are the head server on duty this morning, and it is your job to decide which minutes, if any, will be special, and which pancakes will move where. That is, every minute, you can decide to either do nothing and let the diners eat, or declare a special minute and interrupt the diners to make a single movement of one or more pancakes, as described above.

Breakfast ends when there are no more pancakes left to eat. How quickly can you make that happen?

Input Specification

The first line of the input gives the number of test cases, T. T test cases follow. Each consists of one line with D, the number of diners with non-empty plates, followed by another line with D space-separated integers representing the numbers of pancakes on those diners' plates.

Output Specification

For each test case, output one line containing `Case #x: y`, where x is the test case number (starting from 1) and y is the smallest number of minutes needed to finish the breakfast.

Limits

Memory limit: 1 GB.

$1 \leq T \leq 100$.

Small dataset

Time limit: 30 seconds.

$1 \leq D \leq 6$.

$1 \leq P_i \leq 9$.

Large dataset

Time limit: 60 seconds.

$1 \leq D \leq 1\,000$.

$1 \leq P_i \leq 1\,000$.

Sample Input

```
3
1
3
4
1 2 1 2
1
4
```

Sample Output

```
Case #1: 3
Case #2: 2
Case #3: 3
```

In Case #1, one diner starts with 3 pancakes and everyone else's plate is empty. One optimal strategy is:

Minute 1: Do nothing. The diner will eat one pancake.

Minute 2 (special): Interrupt and move one pancake from that diner's stack onto another diner's empty plate. (Remember that there are always infinitely many diners with empty plates available, no matter how many diners start off with pancakes.) No pancakes are eaten during an interruption.

Minute 3: Do nothing. Each of those two diners will eat one of the last two remaining pancakes.

In Case #2, it is optimal to let the diners eat for 2 minutes, with no interruptions, during which time they will finish all the pancakes.

In Case #3, one diner starts with 4 pancakes and everyone else's plate is empty. It is optimal to use the first minute as a special minute to move two pancakes from the diner's plate to another diner's empty plate, and then do nothing and let the diners eat for the second and third minutes.

Note

This problem has different time limits for different batches. If you exceed the Time Limit for any batch, the judge will incorrectly display `>60.000s` regardless of the actual time taken. Refer to the **Limits** section for batch-specific time limits.

This problem originally had a much higher time limit. However, as reference solutions were much faster, the Time Limit was been reduced accordingly.