Student Placement

Filename: *student*
Time Limit: *1 second*

This year, COP 4516 is bigger than ever! As a consequence, during Friday contests, students sit in four separate locations (HEC-202, outside HEC-208, HEC-308 and outside HEC-308). Each location has a capacity and there are a total number of students in the class. Arup likes maintaining flexibility, so after assigning students to locations, he would like for each location to have extra empty seats, if possible. In particular, he wants to maximize the number, *m*, where *m* represents the minimum of the extra empty seats in any location.

For example, if the capacities of the four locations for students are 6, 10, 25 and 15, and Arup places 3, 2, 20 and 10 students in each of these rooms, respectively, then the rooms have 3, 8, 5 and 5 free seats, respectively. The minimum of these numbers is 3. A better arrangement of the 35 students in the class for this example would be to place 1 in the first room, 5 in the second room, 19 in the third room and 10 in the last room. In this arrangement, the minimum number of empty seats in any one room is 5. (Note that 6 is impossible in this case.)

Unfortunately, Arup is having a bit of problem with the mathematics involved with his query. Write a program to help him determine the maximum number of empty seats he can guarantee in each room, if he places the students in the class appropriately amongst the different contest locations.

The Problem

Given the number of contest locations, the capacity of those locations, and the total number of students in the class, determine the maximum number of empty seats Arup can guarantee in each room, if he places students appropriately.

The Input

The first line of input will consist of a single positive integer, c ($c \leq 25$), representing the number of input cases to process. The first line of each input case consists of two positive integers separated by a space: n ($2 \leq n \leq 10$), representing the number of contest locations and, s ($1 \leq s \leq 10^6$), the number of students enrolled in COP 4516 for the input case. The second line of each input case will contain n space separated positive integers, $k_1, k_2, \ldots, k_n$, where k_i represents the capacity of contest location i ($1 \leq i \leq n$). It is guaranteed that $10^9 \geq k_1 + k_2 + \ldots + k_n \geq s$.

Partial Credit Input (40%)

The number of contest locations, n will equal 2. The number of students in the class, s will be bounded by $1 \leq s \leq 100$. All other bounds are the same.

The Output

For each input case, output the maximum number of empty seats Arup can guarantee in each room, if he places students appropriately, on a line by itself.

Sample Input

```
2
4 35
6 10 25 15
2 57
20 40
```

Sample Output

```
5
1
```