

Kick Start 2021 - Round G

Banana Bunches

Problem

Barbara goes to Alan's banana farm, where the N banana trees are organized in one long line represented by an array \mathbf{B} . The tree at position i has \mathbf{B}_i banana bunches. Each tree has the same cost. Once Barbara buys a tree, she gets all the banana bunches on that tree. Alan has a special rule: because he does not want too many gaps in his line, he allows Barbara to buy at most 2 contiguous sections of his banana tree line.

Barbara wants to buy some number of trees such that the total number of banana bunches on these purchased trees equals the capacity \mathbf{K} of her basket. She wants to do this while spending as little money as possible. How many trees should she buy?

Input

The first line of the input gives the number of test cases, \mathbf{T} . \mathbf{T} test cases follow. Each test case begins with a line containing two integers integer \mathbf{N} , the number of trees on Alan's farm, and \mathbf{K} , the capacity of Barbara's basket. The next line contains \mathbf{N} non-negative integers $\mathbf{B}_1, \mathbf{B}_2, \dots, \mathbf{B}_N$ representing array \mathbf{B} , where the i -th integer represents the number of banana bunches on the i -th tree on Alan's farm.

Output

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 minimum number of trees Barbara must purchase to obtain \mathbf{K} banana bunches using at most 2 contiguous sections of the farm, or -1 if it is impossible to do so.

Limits

Time limit: 20 seconds.

Memory limit: 1 GB.

$1 \leq \mathbf{T} \leq 100$.

$0 \leq \mathbf{B}_i \leq \mathbf{K}$, for each i from 1 to \mathbf{N} .

Test Set 1

$1 \leq \mathbf{K} \leq 10^4$.

$1 \leq \mathbf{N} \leq 50$.

Test Set 2

$1 \leq \mathbf{K} \leq 10^4$.

$1 \leq \mathbf{N} \leq 500$.

Test Set 3

$1 \leq \mathbf{K} \leq 10^6$.

For at most 25 cases:

$$1 \leq N \leq 5000.$$

For the remaining cases:

$$1 \leq N \leq 500.$$

Sample

Sample Input

```
4
6 8
1 2 3 1 2 3
4 10
6 7 5 2
6 8
3 1 2 1 3 1
4 6
3 1 2 0
```

Sample Output

```
Case #1: 3
Case #2: -1
Case #3: 4
Case #4: 3
```

In Sample Case #1, the first section can contain the trees at indices 2 and 3, and the second section can contain the tree at index 6.

In Sample Case #2, it is impossible to achieve a sum of 10 with 2 contiguous sections.

In Sample Case #3, the first section can contain the trees at indices $\{1, 2\}$, and the second section can contain the trees at indices $\{5, 6\}$. We cannot take the $2 + 3 + 3$ combo (trees at indices $\{1, 3, 5\}$) since that would be 3 contiguous sections.

In Sample Case #4, the only section contains the trees at indices $\{1, 2, 3\}$.