</> Problem of the Week 2

Deck of Cards

You are given a deck of custom made cards, denoted by going from the top of the deck to the bottom. Each card has a number which represents its value. You play a strange game with your younger brother, where he tells you his favourite number and you need to find cards and such that and . Since you are older than your brother, you know that finding such a subset of the deck won't always be possible. Thus, you want to write a program which finds two cards and such that the sum is as close as possible to . If there are multiple candidates for the solution, find the one which is lexicographically smallest.

Input

The first line of the input contains the number of test cases. Each of the test cases is described as follows.

- It starts with a line that contains two integers `n k`, separated by a space, where denotes the number of cards, and such that and .
- The following line defines the values of the cards to , in that order. It contains integers , separated by a space, and such that , for . It is guaranteed that .

Output

A solution is a pair of cards with . We define the value of the solution as

For each test case output a single line containing two numbers `i` and `j`, separated by a space, corresponding to the solution with the smallest value. If there are multiple such solutions, output the lexicographically smallest one.

Note: is lexicographically smaller than iff or and .

Points

There are three groups of test sets. For each group there is also a corresponding hidden test set. Overall, you can achieve points.

1. For the first group of test sets, worth points, and the corresponding hidden test set, worth points, you may assume .
2. For the second group of test sets, worth points, and the corresponding hidden test set, worth points, you may assume .
3. For the third group of test test sets, worth points, and the corresponding hidden test set, worth points, there are no additional assumptions.

Corresponding sample test sets are contained in `testi.in/out`, for .