International Olympiad in Informatics 2015
26th July - 2nd August 2015
Almaty, Kazakhstan
Practice session

Search

There is sorted in increasing order sequence A of N distinct integers `A[0]`, `A[1]`, ..., `A[N - 1]`. You are given number N, integer number X and a method allowing you to compare any element of the sequence to any number. Your task is to find position of number X in sequence A.

Example

This example has $N = 6$, $X = 23$.

A

positions	0	1	2	3	4	5
values	3	4	7	9	23	110

If we call function `compare`$(2,20)$ we get 1, because $A_2 = 7 < 20$.
If we call function `compare`$(5,44)$ we get -1, because $A_5 = 110 > 44$.
And finally if we call function `compare`$(4,23)$ we get 0. So the answer is 4.
Note that here we call function only 3 times.

Task

Please write a program that finds position of number X in sequence A.

- `find(sub, N, X)`
  - `sub`: subtask number. $1 \le sub \le 2$.
  - `N`: amount of number in sequence A. $1 \le N \le 100$.
  - `X`: number which you should find. $0 \le X \le 1000$.
  - The function should return positions of number X in sequence A or -1 if there is no such position.

You may call a function `compare(i, val)` that compares number A_i and integer number val. This function returns:

- -1 : if $A_i > val$
- 1 : if $A_i < val$
- 0 : if $A_i = val$

Subtasks

For all subtasks $0 \leq A_i \leq 1000$, $0 \leq i \leq N$. The number of queries is limited. The limit varies by subtask. Your program will receive "wrong answer" if the number of queries exceeds the limit.

subtask	points	number of queries
1	33	100
2	67	7

Implementation details

On each run you program will be given up to 100 test cases.

You have to submit exactly one file, called `search.c`, `search.cpp`, `search.pas` or `search.java`. This file should implement the subprogram described above, using the following signatures.

See provided sample implementation for details on how to access function `compare`.

C/C++ program (include `search.h` at the top of the source file)

```
int find(int sub, int N, int X);
```

Pascal programs (implement the described method in unit `search`, use module `searchlib` to access the API)

```
function find(sub, N, X : longint) : longint;
```

Java programs (implement the described method in public class `search`)

```
int find(int sub, int N, int X);
```

Sample grader

The sample grader reads the input in the following format:

- line 1: `sub T` --- the number of testcases in the file
- line 2: `N X`
- line 3: `A[0]`, ..., `A[N - 1]`
- The next test cases follow. They are given in the same format as the first test case.

The sample grader will print the return value of each `search`.