Trova la permutazione (`permutazione2`)

Dario, Ermanno and Fabrizio have won the ICPC World Finals by solving all the problems two hours before the end. Since they can't exit the hall until the end of the competition, they decide to play the game "Find the permutation" against Davide, Taulant and Tommaso.
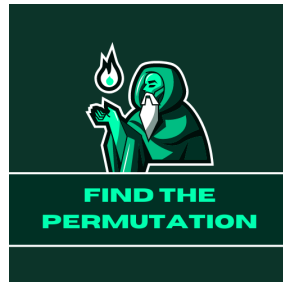

Figure 1: The icon of "Find the permutation"

The game consists in finding a permutation $P_0, P_1, \ldots, P_{N-1}$ of the integers from 0 to $N - 1$. The permutation is hidden, and the players can get information about it only by using a scanner. If the scanner is in position x, it checks the prefix $[0, x]$ of the permutation. Moreover, at every moment the scanner has a state: either L or R.

Initially, the position is $x = 0$ and the state is R. Dario, Ermanno and Fabrizio can ask 3 types of queries:

1. Dario can choose an integer K and ask if K exists in the prefix $[0, x]$ of the permutation (x is the current position of the scanner).

2. Ermanno can change the state of the scanner: if the state was L before the query, it becomes R (and vice versa).

3. Fabrizio can move the scanner. In particular, if the state is L, x decreases by 1; if the state is R, x increases by 1. After each query, x must be an integer between 0 and $N - 1$.

Davide, Taulant e Tommaso have already played the game, asking A, B, C queries of type 1, 2, 3, respectively. The team composed by Dario, Ermanno and Fabrizio wins if they can find the permutation and, for each type of query, they ask at most as many queries as the opponents. Help the winners of the ICPC World Finals to win also this challenge!

Implementation

You should submit a single file, with a `.cpp` extension.

> ☞ Among the attachments in this task you will find a template `permutazione2.cpp` with a sample implementation.

You will have to implement the following function:

```
C++   void indovina(int N, int A, int B, int C, int H[]);
```

- The integer N represents the length of the permutation to find.

- The integer A represents the maximum number of queries of type 1.
- The integer B represents the maximum number of queries of type 2.
- The integer C represents the maximum number of queries of type 3.
- The array H is indexed from 0 to $N-1$ and initialized to 0.
- At the end of the execution of the function, the array H should contain the hidden permutation.

Your program can use the following functions, defined in the grader:

C++	`bool chiedi(int K);`

- This function sends the integer K to the scanner.
- K must be an integer between 0 and $N-1$, otherwise the program will terminate with the message `Invalid query`.
- You can use this function at most A times, otherwise the program will terminate with the message `Too many queries (1)`.
- The function returns `true` if K is located in the prefix $[0, x]$ of P, otherwise it returns `false` (x is the current position of the scanner).

C++	`void stato();`

- This function changes the state of the scanner: if the state was `L` before the query, it becomes `R` (and vice versa).
- You can use this function at most B times, otherwise the program will terminate with the message `Too many queries (2)`.

C++	`void sposta();`

- This function changes the position x of the scanner. In particular, if the state is `L`, x decreases by 1; if the state is `R`, x increases by 1.
- At the end of the execution of the function, the position x must be an integer between 0 and $N-1$, otherwise the program will terminate with the message `Invalid position`.
- You can use this function at most C times, otherwise the program will terminate with the message `Too many queries (3)`.

Sample Grader

Among this task's attachments you will find a simplified version of the grader used during the evaluation, which you can use to test your solutions locally. The sample grader reads data from `stdin`, calls the functions that you should implement and writes back on `stdout` using the following format.

The input file consists of 2 lines, containing:

- Line 1: the integers N, A, B, C.
- Line 2: the array P to guess.

The output file consists of 2 lines:

- Line 1: the array H at the end of the execution of the function `indovina`.
- Line 2: `Correct answer:  ([a], [b], [c]) queries executed` if the array H is equal to the array P, `Wrong answer:  ([a], [b], [c]) queries executed` otherwise. a, b, c are the number of queries of type 1, 2, 3, respectively.

Constraints

- Check the "Scoring" section.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- **Subtask 1** [**0 points**]: Sample cases.
- **Subtask 2** [**19 points**]: $N = 100, \quad A = 10000, \quad B = 10000, \quad C = 10000$
- **Subtask 3** [**12 points**]: $N = 100, \quad A = 6000, \quad B = 6000, \quad C = 6000$
- **Subtask 4** [**12 points**]: $N = 100, \quad A = 3000, \quad B = 3000, \quad C = 3000$
- **Subtask 5** [**25 points**]: $N = 100, \quad A = 1000, \quad B = 1000, \quad C = 10000$
- **Subtask 6** [**16 points**]: $N = 1000, \quad A = 10000, \quad B = 10, \quad C = 10000$
- **Subtask 7** [**16 points**]: $N = 1000, \quad A = 40000, \quad B = 1, \quad C = 2000$

Examples

stdin	stdout
9 1000 1000 1000 3 2 7 8 6 0 5 4 1	3 2 7 8 6 0 5 4 1 Correct answer: (2, 1, 3) queries executed

Explanations

We will show a possible interaction that solves the **first example**:

- Initially, the position is $x = 0$ and the state is R.
- Fabrizio moves the scanner (function `sposta()`). Since the state is R, the position x increases by 1 and it becomes 1.
- Fabrizio moves the scanner again (function `sposta()`). Since the state is R, the position x increases by 1 and it becomes 2.
- Dario asks if 5 is located in the prefix $[0, x]$ of P (function `chiedi(5)`), and he receives the answer `false`.
- Ermanno changes the state of the scanner (function `stato()`). The new state is L.
- Fabrizio moves the scanner (function `sposta()`). Since the state is L, the position x decreases by 1 and it becomes 1.
- Dario asks if 3 is located in the prefix $[0, x]$ of P (function `chiedi(3)`), and he receives the answer `true`.
- Although Dario, Ermanno and Fabrizio don't have enough information to determine the permutation uniquely, they try to guess it. Since the permutation they sent (represented by the array H at the end of the execution of the function) is equal to the hidden one, the testcase has been solved. 2, 1, 3 queries of type 1, 2, 3 respectively have been asked.