



Ancient Machine 2

Bitaro and Bibako are archaeologists who excavated and investigated the ruin of JOI Kingdom. In the ruin, Bitaro found an old stone slate, and Bibako found an old machine.

From the results of the study, Bitaro found out that a string S of length N was written on the stone slate. Each character of the string is either '0' or '1'. However, he does not yet know each character of the string S .

On the other hand, from the results of the study, Bibako found out how to use the machine. To use it, we put the stone slate on the machine, input an integer m and two sequences of integers a, b , and make a query. Here the integer m and the sequences of integers a, b should satisfy the following conditions:

- The integer m is between 1 and 1 002, inclusive.
- Both of the lengths of the sequences a, b are equal to m .
- Every element of the sequences a, b is an integer between 0 and $m - 1$, inclusive.

If we put the stone slate on the machine, input an integer m and two sequences of integers a, b , and make a query, the machine will operate as follows and will show an integer.

1. The machine sets 0 in the **memory area** of the machine.
2. The machine performs the following N **operations**. The $(i + 1)$ -th ($0 \leq i \leq N - 1$) operation proceeds as follows.

Let x be the current integer set in the memory area of the machine. The machine reads the character S_i . Here, S_i is the i -th character of the string S , if we count the characters of the string S so that the first character is the 0-th character.

- If S_i is '0', the machine sets a_x in the memory area of the machine.
Here, a_x is the x -th element of the sequence a , if we count the elements of the sequence a so that the first element is the 0-th element.
- If S_i is '1', the machine sets b_x in the memory area of the machine.
Here, b_x is the x -th element of the sequence b , if we count the elements of the sequence b so that the first element is the 0-th element.

3. The machine shows the integer which is finally set in the memory area.

Using the machine, Bitaro wants to specify the string written on the stone slate. However, since the machine is very fragile, the number of queries cannot exceed 1 000. Moreover, the maximum of the integer m input to the machine for a query should be as small as possible.

Write a program which, using the machine, specifies the string written on the stone slate.



Implementation Details

You need to submit one file.

The file is `ancient2.cpp`. It should implement the following function. The program should include `ancient2.h` using the preprocessing directive `#include`.

- `std::string Solve(int N)`

This function is called only once for each test case. This function should return a string which is the same as the string S written on the stone slate.

- The parameter N is the length N of the string S written on the stone slate.
- This function should return a string of length N . If the length is different from N , your program is judged as **Wrong Answer [1]**.
- Each character of the return value is either '0' or '1'. If this condition is not satisfied, your program is judged as **Wrong Answer [2]**.
- The return value should be the same as the string S written on the stone slate. If this condition is not satisfied, your program is judged as **Wrong Answer [3]**.

Your program can call the following function.

- ★ `int Query(int m, std::vector<int> a, std::vector<int> b)`

Using this function, your program can make a query.

- ◊ The parameter m is the integer m input to the machine.
- ◊ The parameters a, b are the two sequences of integers a, b input to the machine.
- ◊ The return value is the integer shown by the machine when we put the stone slate on the machine, input the above integer and sequences, and make a query.
- ◊ The parameter m should be between 1 and 1 002, inclusive. If this condition is not satisfied, your program is judged as **Wrong Answer [4]**.
- ◊ Both of the lengths of the parameters a, b are equal to m . If this condition is not satisfied, your program is judged as **Wrong Answer [5]**.
- ◊ Every element of the parameters a, b is between 0 and $m - 1$, inclusive. If this condition is not satisfied, your program is judged as **Wrong Answer [6]**.
- ◊ The function `Query` should not be called more than 1 000 times. If it is called more than 1 000 times, your program is judged as **Wrong Answer [7]**.



Important Notices

- Your program can implement other functions for internal use, or use global variables.
- Your program must not use the standard input and the standard output. Your program must not communicate with other files by any methods. However, your program may output debugging information to the standard error.

Compilation and Test Run

You can download an archive file from the contest webpage which contains the sample grader to test your program. The archive file also contains a sample source file of your program.

The sample grader is the file `grader.cpp`. In order to test your program, put `grader.cpp`, `ancient2.cpp`, `ancient2.h` in the same directory, and run the following command to compile your programs. Instead, you may run `compile.sh` contained in the archive file.

```
g++ -std=gnu++17 -O2 -o grader grader.cpp ancient2.cpp
```

When the compilation succeeds, the executable file `grader` is generated.

Note that the actual grader is different from the sample grader. The sample grader will be executed as a single process, which will read input data from the standard input and write the results to the standard output.

Input for the Sample Grader

The sample grader reads the following data from the standard input.

```
N
S
```

Output of the Sample Grader

The sample grader outputs the following information to the standard output (quotes for clarity).

- If your program is judged as correct, it reports the maximum of the parameter `m` of the function `Query` as “Accepted: 22”. However, if your program is judged as correct without calling the function `Query`, it outputs “Accepted: 0”.
- If your program is judged as any type of Wrong Answer, the sample grader writes its type as “Wrong



Answer [4]”.

If your program satisfies the conditions of several types of Wrong Answer, the sample grader reports only one of them.

Constraints

All the input data satisfy the following conditions.

- $N = 1\,000$.
- S is a string of length N .
- Each character of the string S is either ‘0’ or ‘1’.

Notices for Grading

For every test case, the actual grader is **not** adaptive. This means the grader has a fixed answer in the beginning.

Grading

If your program is judged as Wrong Answer for any of the test cases, your score for this task is 0.

If your program is judged as correct for all the test cases, your score is calculated by M , which is the maximum of the parameter m of the function `Query` for all the test cases. However, if your program is judged as correct without calling the function `Query`, your score is calculated by $M = 0$.

- If $103 \leq M \leq 1\,002$, your score is $10 + \left\lfloor \frac{(1002-M)^2}{9000} \right\rfloor$ points.
- If $0 \leq M \leq 102$, your score is 100 points.

Here $\lfloor x \rfloor$ is the largest integer not exceeding x .



Sample Communication

Here is a sample input for the sample grader and corresponding function calls.

Sample Input 1	Sample Function Calls			
	Function Call to Solve	Return Values	Function Call to Query	Return Values
3 110	Solve(3)			
			Query(4, [3, 3, 2, 2], [2, 2, 1, 0])	3
			Query(2, [0, 1], [1, 0])	0
			Query(1, [0], [0])	0
			Query(3, [1, 1, 1], [1, 1, 1])	1
		"110"		

Assume that the string S written on the stone slate is "110". If we put the stone slate on the machine, input $(m, a, b) = (4, [3, 3, 2, 2], [2, 2, 1, 0])$, and make a query, the machine will operate as follows.

1. The machine sets 0 in the memory area of the machine.
2. For the first operation, since S_0 is '1', the machine sets b_0 , i.e. 2, in the memory area of the machine.
3. For the second operation, since S_1 is '1', the machine sets b_2 , i.e. 1, in the memory area of the machine.
4. For the third operation, since S_2 is '0', the machine sets a_1 , i.e. 3, in the memory area of the machine.
5. Since the integer which is finally set in the memory area is 3, the machine shows 3.

Note that this sample input **does not satisfy** the constraint $N = 1\,000$.

Among the files which can be obtained from the contest webpage, `sample-02.txt` satisfies the constraint.