# #@ck-the-OJ

**Project**
Data Structures
Algorithms
Due date: 3 May, 2020

**Problem Statement:** You realise that soon you will be graduating(to sophomore year) and will miss OJ badly, in spite of such a pain in the ass(for instance, the last question). So you decide to leave your final impression. If the normal assignments can be buggy, imagine the level of chaos in the interactive question! A graduating senior informed about the bug in the code which lets you exploit a special PIN which gives you the ADMIN access to OJ. The PIN has following properties:

1. It has a fixed length of 4 digits.
2. All the digits are distinct.
3. There can be a leading 0 in the PIN.

The bug lets you exploit that pin. It is invoked by "TIK" command and if you correctly guessed the PIN, you can destroy the entire OJ by using command "TOK <correct PIN >"(imagine the amount of blessings you will get!). The bug works in the following manner:

1. Provide it with "TIK <your guessed PIN >"
2. It returns two digits C and D.
3. C indicates how many digits are placed at its correct position and D indicates how many wrong digits are provided by you in the guessed PIN.

The correct guessed PIN would return 4 0. (You might wonder why can't it give the correct PIN itself. That's because it's constantly under the watch of Big Brother, and explicitly providing the PIN will alert Big B). However there are many catches:

1. You cannot ask more than 7 TIK queries, or else Big Brother will know.
2. Your have to do this correctly Q times in order to bring down the OJ, and each time the PIN will change.

**Note**

BE EXTRA SURE OF THE INTERACTION. IT IS CASE SENSITIVE AND SPACE SENSITIVE. DO NOT PRINT ANY EXTRA LINES AND DO NOT PRINT ANY OTHER DEBUGGING STATEMENTS. THIS QUESTION IS NOT JUST ABOUT LOGIC, BUT OPTIMIZATION AS WELL.

### Input

First fixed line will contain Q, the number of times you have to guess the PIN correctly.

### Interaction

Query "TIK <guessed PIN >"(without quotes), and the bug will return two space separated properties C and D of the guessed PIN as mentioned above. Note that you don't need to give PIN with leading zeros, and value returned can be directly inputted as an integer. NOTE: AFTER USING cout, ALWAYS USE fflush(NULL); statement in C++. If you don't use that, there will be issues with input.

**Output**

Once you find the correct PIN, output ”TOK <correct PIN >”(without quotes).

**Constraints**

$1 \leq Q \leq 3000$<br>
$0 \leq C, D \leq 4$<br>
special TIK queries $\leq 7$

**Time Limit**: 10 sec<br>
**Memory Limit**: 256 MB

**Sample Test Case**

| | |
|---|---|
| PINS | 0123<br>2015<br>1053 |
| Interaction Sample 1(on the above arrays) | |
| 3 | TIK 123 |
| 4 0 | TOK 123 |
| 0 1 | TIK 123 |
| 1 0 | TIK 125 |
| 1 0 | TIK 1052 |
| 1 0 | TIK 2501 |
| 2 0 | TIK 2510 |
| 2 0 | TIK 1025 |
| 4 0 | TIK 2015 |
| 1 1 | TOK 2015 |
| 0 1 | TIK 123 |
| 2 0 | TIK 125 |
| 4 0 | TIK 1035 |
| | TIK 1053 |
| | TOK 1053 |

**Explanation**

The file has 3 codes to be broken as given above. There is no fixed input, i.e. copy pasting the left side of table to test your code won't work.

For the first code, querying 123 means querying 0123, and that compared with 0123 gives 4 places where digits are correctly placed and no wrongly guessed digits.

For the second code, querying 123 means querying 0123, and that compared with 2015 gives 0 places where digits are correctly placed and 1 wrongly guessed digits(3 is wrongly guessed instead of 5). Similarly, the other queries responds to the various permutation of 2, 0, 1 and 5.

For the third code, querying 123 means querying 0123, and that compared with 1053 gives 1 place where digits are correctly placed(the last 3) and 1 wrongly guessed digits(3 is wrongly guessed instead of 5). Similarly, querying 125 means querying 0125, and that compared with 1053 gives 0 places where digits are correctly placed and 1 wrongly guessed digits(2 is wrongly guessed instead of 3).