NZPC

New Zealand Programming Contest

PRACTICE PROBLEM SET

This problem set is to be used during the practice contest, whose purpose is to ensure that DOMjudge is working correctly, and to make sure teams know how to submit solutions.

It is a good idea to submit some deliberately wrong answers to check out the error messages, and to submit at least one clarification.

NZPC
New Zealand
Programming Contest

PREAMBLE

Please note the following very important details relating to input and output:

- Read all input from the keyboard, i.e. use `stdin`, `System.in`, `cin`, `Console.ReadLine` or equivalent. Input will be redirected from a file to form the input to your submission.
- Do NOT prompt for input as this will appear in your output and cause a submission to be judged as wrong.
- Write all output to the screen, i.e. use `stdout`, `System.out`, `cout`, `Console.WriteLine` or equivalent. Do not write to `stderr`.
- Unless otherwise stated, all *integers* will fit into a standard 32-bit computer word. If more than one integer appears on a line, they will be separated by a space.
- An *uppercase letter* is a character in the sequence 'A' to 'Z'. A *lower case letter* is a character in the sequence 'a' to 'z'. A *letter* is either a lower case letter or an upper case letter.
- Unless otherwise stated, a *name* is a continuous sequence of from 2 to 30 characters (printed or written letters or symbols).
- If it is stated that 'a line contains no more than *N* characters', this does not include the character(s) specifying the end of line.
- Input files are sometimes terminated by a 'sentinel' line. This line should not be processed.

Please also note that:

- The filenames of your submitted programs may need to follow a particular naming convention, for example the name of a Java file containing a public class needs to be the name of the class followed by the '.java' extension.
- DOMjudge will reject a submitted file which has any spaces in its file name.
- Unless otherwise specified, all problems have a time limit of 1 second. A *TIMELIMIT* error will be issued for submissions that exceed that limit on a single test run.
- Each problem description in the main contest takes up 2 pages, one of which may be empty.

PRACTICE PROBLEM A **HELLO NZPC** **1 POINT**

This problem doesn't read input. You simply need to output the single line shown below.

Input

There is no input.

Output

Output consists of the single line:

```
Welcome! This is the NZ Programming Contest.
```

Sample Input 1	Output for Sample Input 1
	`Welcome! This is the NZ Programming Contest.`

Note

You may like to check what happens when you submit an answer with one of the punctuation marks missing, one with a lower case 'w', and one with completely wrong text eg "Fried potatoes."

PRACTICE PROBLEM B **JELLY BEANS** **3 POINTS**

A popular event at fairs and fundraisers is “How many jelly beans are in the jar?” Participants guess the number and the one who comes closest wins the jar. If two people are equally close, the first one who guessed wins the prize.

Input

The first line of input is the number of jelly beans in the jar. This will be between 50 and 1,000 inclusive.

The second line is the number of people who had a guess, a number between 1 and 250 inclusive.

The guesses then follow, two lines per guess.

- The first line of each guess is the name of the person who guessed. Each name is a single word as defined in the preamble. Names are not repeated.
- The second line is the person’s guess, a positive integer.

Output

Output consists of a single line containing just the name of the winner.

Sample Input

```
256
8
Alan
240
Betty
232
Cheng
253
Dimitri
284
Evan
300
Fathima
258
Gurtrude
206
Harry
238
```

Output for Sample Input

```
Fathima
```

Explanation

Fathima’s guess was just 2 away from the correct answer, beating Cheng who was 3 away.

NZPC
New Zealand Programming Contest

PROBLEM C **ATM TRANSACTIONS** **3 POINTS**

Whenever somebody goes to an ATM to withdraw or deposit money, a calculation has to be done to keep the person's bank balance correct. Your task in this problem is to do such calculations.

There is a bank rule that says that a customer may not have an overdraft of more than $200, so any withdrawal that would take the balance below -200 must be stopped. (A minus sign is used to indicate an overdraft, or negative balance).

Input

Input consists of up to 100 lines, each representing a transaction.

Each transaction consists of an integer representing the starting balance (between -200 and +10,000 inclusive), the letter W or the letter D (Withdraw or Deposit), followed by a second integer representing the amount to be withdrawn or deposited (between 5 and 400 inclusive).

Input will be terminated by a line containing `0 W 0`. Do not process this line.

Output

Output consists of one line for each line of input showing the new balance after each transaction, except that if a withdrawal would take the balance below -200, the output must be the words `Not allowed`.

Sample Input

```
100 W 10
-200 D 300
50 W 300
0 W 0
```

Output for Sample Input

```
90
100
Not allowed
```