ACM International Collegiate Programming Contest

2011

Latin American Regional Contests

November 4th-5th, 2011

Warmup Session

This problem set contains 2 problems; pages are numbered from 1 to 2.

This problem set is used in simultaneous contests hosted in the following countries:

- Argentina
- Bolivia
- Brazil
- Chile
- Colombia
- Cuba
- Peru
- Mexico
- Venezuela

General Information

Unless otherwise stated, the following conditions hold for all problems.

Input

1. The input must be read from standard input.
2. The input contains several test cases. Each test case is described using a number of lines that depends on the problem.
3. When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input. There are no empty lines.
4. Every line, including the last one, has the usual end-of-line mark.
5. The end of input is indicated with a line containing certain values that depend on the problem. This line should not be processed as a test case.

Output

1. The output must be written to standard output.
2. The result of each test case must appear in the output using a number of lines that depends on the problem.
3. When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output. There should be no empty lines.
4. Every line, including the last one, must have the usual end-of-line mark.
5. No special mark should be written to indicate the end of output.

Problem A

Bakugan

Problem code name: `bakugan`

Mark and Leti love to play with Bakugan balls. These balls are small plastic spheres with a tiny monster toy inside. When dropped to the ground, a Bakugan ball pops open with an incredible sound, liberating a fearsome Bakugan monster. Mark and Leti love to play with the toy monsters, but popping open the balls is also great fun.

Each of them received a bag with Bakugan balls, and they invented a game to pop open the balls. There are 10 different monsters, and for the game Mark and Leti associated each monster with a different integer from 1 to 10, according to the monster's ugliness. The game is composed of R rounds. At each round:

- both players drop simultaneously a ball each;
- each player accumulates a number of points coincident with the number associated with the monster liberated by her/his ball;
- the first (and only the first) player who liberates the same monster in three consecutive rounds earns 30 additional points; if this condition happens in the same round for both players then nobody earns extra points.

The winner of the game is the player who accumulates more points. Please help Mark and Leti announce the winner of the game!

Input

Each test case is described using three lines. The first line contains an integer R representing the number of rounds of the game ($1 \leq R \leq 10$). The second line contains R integers M_i indicating the monsters liberated by Mark at each turn ($1 \leq M_i \leq 10$ for $1 \leq i \leq R$). The third line contains R integers L_i indicating the monsters liberated by Leti at each turn ($1 \leq L_i \leq 10$ for $1 \leq i \leq R$).

The last test case is followed by a line containing one zero.

Output

For each test case output a line with a character representing the result of the game: '`M`' (uppercase) if the winner is Mark, '`L`' (uppercase) if the winner is Leti, or '`T`' (uppercase) if there is a tie.

Sample input	**Output for the sample input**

`10`
`4 2 2 2 5 6 7 8 1 1`
`1 4 4 4 1 1 1 1 2 3`
`5`
`3 3 3 3 2`
`8 9 9 9 9`
`10`
`8 4 7 1 1 9 5 2 4 3`
`5 6 9 7 9 4 2 3 7 4`
`0`
	`M`
	`T`
	`L`

Problem B

Hailstone Sequences

Problem code name: `hailstone`

Consider the sequence formed by starting from a positive integer h_0 and iterating with $n = 1, 2, \ldots$ the following definition until $h_n = 1$:

$$h_n = \begin{cases} \frac{1}{2} \times h_{n-1} & \text{if } h_{n-1} \text{ is even;} \\ \\ 3 \times h_{n-1} + 1 & \text{if } h_{n-1} \text{ is odd.} \end{cases}$$

For instance, if we start with $h_0 = 5$ the following sequence is generated: 5, 16, 8, 4, 2, 1. If we start with $h_0 = 11$, the sequence generated is 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1.

As you can see from these examples, the numbers go up and down, but eventually comes down to end in 1 (at least for all numbers that have ever been tried). These sequences are called *Hailstone* sequences because they are similar to the formation of hailstones, which get carried upward by the winds over and over again before they finally descend to the ground.

In this problem, given a positive integer, your task is to compute the highest number in the Hailstone sequence which starts with the given number.

Input

Each test case is described using a single line. The line contains an integer H representing the starting value to build the sequence ($1 \leq H \leq 500$).

The last test case is followed by a line containing one zero.

Output

For each test case output a line with an integer representing the highest number in the Hailstone sequence that starts with the given input value.

Sample input	**Output for the sample input**
5 11 27 0	16 52 9232