Problem L. Latin Language Lesson

Time limit: 2 seconds

This is an interactive problem.

On a boring Latin Language lesson, Julius and Octavian are playing the following game. A valid non-empty Roman numeral is written on the board.

The following table from Wikipedia displays how Roman numerals are written:

Individual decimal places	Thousands	Hundreds	Tens	Units
1	M	C	X	I
2	MM	CC	XX	II
3	MMM	CCC	XXX	III
4		CD	XL	IV
5		D	L	V
6		DC	LX	VI
7		DCC	LXX	VII
8		DCCC	LXXX	VIII
9		CM	XC	IX

The composition rule:

- The numerals for 4, 9, 40, 90, 400, and 900 are written using "subtractive notation" where the first symbol is subtracted from the larger one: for example, 40 ("`XL`") is written as '`X`' (10) subtracted from '`L`' (50). These are **the only** subtractive forms in standard use.
- A number containing several decimal digits is built by appending the Roman numeral equivalent for each, from highest to lowest.
- Any missing place (represented by a zero in the place-value equivalent) is omitted.
- The largest number that can be represented in the Roman notation is 3999 (MMMCMXCIX).

Julius can choose who makes the first move. Then the players take turns making moves. A move consists of adding one letter to the left or to the right of the number already written on the board in such a way that the resulting number remains a valid Roman numeral. The player who cannot make the next move loses.

Your task is, given the initial number, to win while playing for Julius, and the computer will be playing for Octavian. You may choose your turn (first or second).

Interaction Protocol

One test consists of several scenarios. First, the jury program prints a line with an integer t, the number of scenarios ($1 \le t \le 100$).

In each scenario, the interaction starts with the jury program printing a line with a valid non-empty Roman numeral. Then the participant chooses the player to move first: the participant themselves (Julius) or the computer (Octavian). To do this, they print a line with a single integer: `1` if the participant makes the first move, or `2` if the computer makes the first move. Then the players take turns making moves, printing the Roman numeral that is obtained after their move. Each move is printed on a single line.

If the number is not a valid Roman numeral, or if the number cannot be obtained by adding one letter to the left or to the right of the previous number, then the corresponding player loses. The jury program will not attempt to make invalid moves. Instead, if the jury program loses, it outputs the text `GG` on a separate line and proceeds to process the next scenario, if any, or terminates otherwise.

The jury program can use various strategies, and its actions may depend on the participant's moves (i.e., the interactor is adaptive).

Example

standard input	*standard output*
2 MMII MMMII GG MMIV GG	 2 MMMIII 1 MMMIV