

## Bubbles and Buckets

*Extracted from: UVa*

*Source file name: buckets.py*

*Time limit: 3*

Andrea, Carlos and Marcelo are close friends and spend their weekends by the swimming pool. While Andrea gets a suntan, both friends play *Bubbles*. Andrea, a very smart computer scientist, has already told them that she does not understand why they spend so much time playing a game so simple.

Using her laptop, Carlos and Marcelo generate a random integer  $N$  and a sequence, also random, which is a permutation from  $1, 2, \dots, N$ .

The game then begins. The players play by turns, and at each turn a player makes a move. Marcelo is always the first to play. A move consists of choosing one pair of consecutive elements that are out of order in the sequence, and swapping both elements. For example, given the sequence  $1, 5, 3, 4, 2$ , a player may swap 3 and 5 or 4 and 2, but cannot swap 3 and 4 nor 5 and 2. Continuing with the example, if the player decides to swap 5 and 3, the new sequence will be  $1, 3, 5, 4, 2$ .

Sooner or later, the sequence will be sorted. The player that cannot make a move loses.

Andrea, with disdain, always says that it would be simpler to play Odd or Even, to the same effect. Your mission, in case you decide to accept it, is to determine who wins the game, given the initial permutation  $P$ .

### Input

The input contains several test cases. Each test case is composed of a single line, in which all integer are separated by one space. Each line contains an integer  $N$  ( $2 \leq N \leq 10^5$ ), followed by the initial sequence  $P = (X_1, X_2, \dots, X_N)$  of  $N$  distinct integers, with  $1 \leq X_i \leq N$  for  $1 \leq i \leq N$ .

The end of input is indicated by a line containing only one zero.

*The input must be read from standard input.*

### Output

For each test case in the input, your program must print a single line, containing the name of the winner, equal to 'Carlos' or 'Marcelo'.

*The output must be written to standard output.*

Sample Input	Sample Output
5 1 5 3 4 2	Marcelo
5 5 1 3 4 2	Carlos
5 1 2 3 4 5	Carlos
6 3 5 2 1 4 6	Carlos
5 5 4 3 2 1	Carlos
6 6 5 4 3 2 1	Marcelo
0	