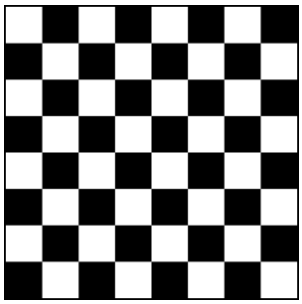
Chess Board

As Mr. Panda is predominantly black and white, he loves to play chess. However, it is very difficult for him to find friends to play chess with him. Thus, he wants to create his own software with his own chess AI so he can play chess all by himself.

Chess is usually played on an 8 by 8 board where each square is either black or white in colour. The squares are coloured such that the top-left square is white and no 2 squares touching each other have the same colour. For example, an 8 by 8 board would be coloured like this:



Mr. Panda is bored of always playing on an 8 by 8 board and wants to try playing on chess boards of different sizes. The first step to coding his software is to be able to generate these different sized chessboards. Can you help him generate these boards?

Input
The input will consist of 1 line with two integers, **R** and **C.**
These 2 numbers represent the number of rows and the number of columns in the chessboard respectively.

Output
Your program needs to output a chessboard with **R** rows and **C** columns.
For each white square, print an 'O' character and for each black square print an 'X' character.
The board should be coloured such that the top-left square is white and no 2 squares touching each other have the same colour.

Limits

- $1 \leq$ **R, C** $\leq 100{,}000$
- **R x C** $\leq 100{,}000$

Sample Input (**chessboard1.in**)	Sample Output (**chessboard1.out**)
8 8	OXOXOXOX XOXOXOXO OXOXOXOX XOXOXOXO OXOXOXOX XOXOXOXO OXOXOXOX XOXOXOXO

Sample Input (**chessboard2.in**)	Sample Output (**chessboard2.out**)
2 4	0X0X X0X0

Notes:

1. You should develop your program in the subdirectory **ex1** and use the skeleton java file provided. You should not create a new file or rename the file provided.
2. You are free to define your own helper methods and classes (or remove existing ones).
3. Please be reminded that the marking scheme is:
   a. Public Test Cases (1%) - 1% for passing **all** test cases, 0% otherwise
   b. Hidden Test Cases (1%) - Partial scoring depending on test cases passed
   c. Manual Grading (1%)
      i. Overall Correctness (correctness of algorithm, severity of bugs)
      ii. Coding Style (meaningful comments, modularity, proper indentation, meaningful method and variable names)
4. Your program will be tested with a time limit of not less than **1 sec** on Codecrunch.

Skeleton File – Chessboard.java

You are given the below skeleton file `Chessboard.java`. You should see a non-empty file when you open the skeleton file. Otherwise, you might be in the wrong working directory.

```
/**
 * Name         :
 * Matric. No   :
 * PLab Acct.   :
 */

import java.util.*;

public class Chessboard {
    private void run() {
        //implement your "main" method here
    }

    public static void main(String[] args) {
        Chessboard newChessboard = new Chessboard();
        newChessboard.run();
    }
}
```