A. Cards

User ainta loves to play with cards. He has a cards containing letter "o" and b cards containing letter "x". He arranges the cards in a row, and calculates the score of the deck by the formula below.

1. At first, the score is 0.
2. For each block of contiguous "o"s with length x the score increases by x^2.
3. For each block of contiguous "x"s with length y the score decreases by y^2.

For example, if $a = 6$, $b = 3$ and ainta have arranged the cards in the order, that is described by string "ooxoooxxo", the score of the deck equals $2^2 - 1^2 + 3^2 - 2^2 + 1^2 = 9$. That is because the deck has 5 blocks in total: "oo", "x", "ooo", "xx", "o".

User ainta likes big numbers, so he wants to maximize the score with the given cards. Help ainta make the score as big as possible. Note, that he has to arrange all his cards.

Input

The first line contains two space-separated integers a and b ($0 \le a, b \le 10^5$; $a + b \ge 1$) — the number of "o" cards and the number of "x" cards.

Output

In the first line print a single integer v — the maximum score that ainta can obtain.

In the second line print $a + b$ characters describing the deck. If the k-th card of the deck contains "o", the k-th character must be "o". If the k-th card of the deck contains "x", the k-th character must be "x". The number of "o" characters must be equal to a, and the number of "x " characters must be equal to b. If there are many ways to maximize v, print any.

Please, do not write the `%lld` specifier to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams or the `%I64d` specifier.

Examples

input
2 3

output
-1
xoxox

input
4 0

output
16
oooo

input
0 4

| output |

```
-16
xxxx
```