Weaving

Time limit: 1 sec

Given two list **a** = [a1, a2, ..., an] and **b** = [b1, b2, ..., bm], produce new list whose members are members of **a** intertwine with the members of **b**. The resulting list maintain the relative order of each list and the members alternate between member of a and member of b whenever possible. For example, assuming that n < m, the result of the above lists are [a1, b1, a2, b2, ...,an,bn,bn+1, bn+2, ..., bm].

Your Task

Write a function called “weaving” in the given code.

Input

Input has exactly two lines, each of both lines contain a list of Int. The length of each list does not exceed 10,000 members

Output

Output exactly one line containing the merged list

Example

Input	Output
[1,2,3] [999,888,777]	[1,999,2,888,3,777]
[10,20,30,40] [-1,-2]	[10,-1,20,-2,30,40]
[] [1,2,3]	[1,2,3]

Haskell Input

Please use the following starting code. The code reads two lists from the keyboard and call the function weaving

```
main :: IO ()
main = do
         x1 <- readLn
         x2 <- readLn
         putStrLn (show (weaving x1 x2))
weaving :: [Int] -> [Int] -> [Int]
-- write your code here
```