Monocarp has got two strings s and t

having equal length. Both strings consist of lowercase Latin letters "a" and "b".

Monocarp wants to make these two strings s

and t equal to each other. He can do the following operation any number of times: choose an index pos_1 in the string s, choose an index pos_2 in the string t, and swap s_{pos_1} with t_{pos_2}

.

You have to determine the minimum number of operations Monocarp has to perform to make s

and t

equal, and print any optimal sequence of operations — or say that it is impossible to make these strings equal.

Input

The first line contains one integer n

$(1 \le n \le 2 \cdot 10^5)$ — the length of s and t

.

The second line contains one string s

consisting of n

characters "a" and "b".

The third line contains one string t

consisting of n

characters "a" and "b".

Output

If it is impossible to make these strings equal, print -1

.

Otherwise, in the first line print k

— the minimum number of operations required to make the strings equal. In each of the next k lines print two integers — the index in the string s and the index in the string t

that should be used in the corresponding swap operation.

Examples

Input	Output
4 abab aabb	2 3 3 3 2

Input	Output
1 a b	-1

Input	Output
8 babbaabb abababaa	3 2 6 1 3 7 8

Note

In the first example two operations are enough. For example, you can swap the third letter in s

with the third letter in t. Then s= "abbb", t= "aaab". Then swap the third letter in s and the second letter in t. Then both s and t

are equal to "abab".

In the second example it's impossible to make two strings equal.