Substrings

An array of characters v_2 with length l_2 is a ***substring*** of another array of characters v_1 with length l_1, $0 < l_2 \leq l_1 \leq 100$, if all the elements in array v_2 are contained in array v_1 in the same order and in consecutive positions.

Our purpose is to develop a recursive algorithm that checks whether v_2 is a substring of v_1 or not.

In addition to coding the solution, you have to specify the algorithm and calculate its complexity. What happens if $l_1/2 < l_2 \leq l_1 \leq 1000$***?***

Input

Each test case is described in three lines. The first one contains the length of arrays v_1 and v_2, the second contains the characters of array v_1 and the third contains the characters of array v_2.

The input ends when both arrays are empty (and it must not be processed).

Output

The output of each case is an independent line saying YES if v_2 is a substring of v_1 and NO otherwise.

Sample input

```
3 1
a b c
a
3 2
a b c
d c
3 1
a b c
b
3 2
a b c
b c
3 2
a b c
a c
3 3
a a c
a a c
0 0
```

Sample output

```
YES
NO
YES
YES
NO
YES
```

Notes

This exercise must be understood in the context of the *Data Structures and Algorithms* course, FDI-UCM 2016/2017 (prof. Gonzalo Méndez). Therefore, the only valid solutions are those that use the concepts studied in this course. Additional remarks may be provided in class.