

Qingshan Loves Strings

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Qingshan has a string s , while Daniel has a string t . Both strings only contain 0 and 1.

A string a of length k is good if and only if

- $a_i \neq a_{i+1}$ for all $i = 1, 2, \dots, k - 1$.

For example, 1, 101, 0101 are good, while 11, 1001, 001100 are not good.

Qingshan wants to make s good. To do this, she can do the following operation any number of times (possibly, zero):

- insert t to any position of s (getting a new s).

Please tell Qingshan if it is possible to make s good.

Input

The input consists of multiple test cases. The first line contains a single integer T ($1 \leq T \leq 2000$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers n and m ($1 \leq n, m \leq 50$) — the length of the strings s and t , respectively.

The second line of each test case contains a string s of length n .

The third line of each test case contains a string t of length m .

It is guaranteed that s and t only contain 0 and 1.

Output

For each test case, print “YES” (without quotes), if it is possible to make s good, and “NO” (without quotes) otherwise.

You can print letters in any case (upper or lower).

Example

standard input	standard output
5	Yes
1 1	Yes
1	No
0	No
3 3	No
111	
010	
3 2	
111	
00	
6 7	
101100	
1010101	
10 2	
1001001000	
10	

Note

In the first test case, s is good initially, so you can get a good s by doing zero operations.

In the second test case, you can do the following two operations (the inserted string t is underlined):

1. 101011
2. 101010101

and get $s = 101010101$, which is good.

In the third test case, there is no way to make s good after any number of operations.