

Хрестики-нулики

Назва вхідного файлу:	<code>standard input</code>
Назва вихідного файлу:	<code>standard output</code>
Ліміт часу:	<code>1 second</code>
Ліміт використання пам'яті:	<code>256 megabytes</code>

Ви грали в хрестики нулики з другом онлайн. Однак сталася халепа, знову повітряна тривога у вашому районі! Ви, як вихована людина, йдете у бомбосховище. По закінченню тривоги Ви приходите догравати гру, але щось тут не так. Ваш друг міг обманути й змінити поле.

Ви запам'ятали поле гри A коли йшли. Повернувшись, ви бачите ту саму гру і поле B . Скажіть, чи можливо отримати з поля A поле B за не більше ніж один крок, зроблений за правилами?

Нагадаємо, що перший крок робить гравець, що ставить **хрестики**. Зверніть увагу, що тут не діє стандартне правило про те, що якщо є три хрестики чи нулика підряд, то гра зупиняється.

Формат вхідних даних

Перші три рядки містять по три символи $A_{i,j}$ — опис початкового поля.

Наступні три рядки містять по три символи $B_{i,j}$ — опис кінцевого поля.

Кожна клітинка поля описується трьома символами:

- «.» — позначає пусту клітинку;
- «0» — позначає поставлений нуль;
- «X» — позначає поставлений хрестик.

Гарантується, що поле A можна отримати послідовністю правильних ходів з пустого поля.

Формат вихідних даних

Виведіть «YES» чи «NO» (у будь-якому регістрі) залежно від того, чи можна було отримати поле B з поля A .

Приклади

standard input	standard output
.X. .X. 00. .X. .X. 000	NO
.XX .00 ... XXX .00 ...	YES
XXX 000 ... XXX 000 .X.	YES
0X0 X.X 0X0 X0X 0.0 X0X	NO
.X.X.	YES

Зауваження

У першому прикладі додається один нулик, проте зараз черга поставити хрестик.

У другому прикладі додається один хрестик.

У третьому прикладі також додається один хрест. Зверніть увагу, що попри те, що є три нулики (і хрестики) підряд — гра не зупиняється.

У четвертому прикладі поле змінене.

У п'ятому прикладі жодних кроків не було зроблено.