

## A. Little Elephant and Chess

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

The Little Elephant loves chess very much.

One day the Little Elephant and his friend decided to play chess. They've got the chess pieces but the board is a problem. They've got an  $8 \times 8$  checkered board, each square is painted either black or white. The Little Elephant and his friend know that a *proper chessboard* doesn't have any side-adjacent cells with the same color and the upper left cell is white. To play chess, they want to make the board they have a proper chessboard. For that the friends can choose any row of the board and cyclically shift the cells of the chosen row, that is, put the last (rightmost) square on the first place in the row and shift the others one position to the right. You can run the described operation **multiple times** (or not run it at all).

For example, if the first line of the board looks like that "BBBBBBWW" (the white cells of the line are marked with character "W", the black cells are marked with character "B"), then after one cyclic shift it will look like that "WBBBBBBW".

Help the Little Elephant and his friend to find out whether they can use any number of the described operations to turn the board they have into a proper chessboard.

### Input

The input consists of exactly eight lines. Each line contains exactly eight characters "W" or "B" without any spaces: the  $j$ -th character in the  $i$ -th line stands for the color of the  $j$ -th cell of the  $i$ -th row of the elephants' board. Character "W" stands for the white color, character "B" stands for the black color.

Consider the rows of the board numbered from 1 to 8 from top to bottom, and the columns — from 1 to 8 from left to right. The given board can initially be a proper chessboard.

### Output

In a single line print "YES" (without the quotes), if we can make the board a proper chessboard and "NO" (without the quotes) otherwise.

### Examples

input
WBWBWBWB BWBWBWBW BWBWBWBW BWBWBWBW WBWBWBWB WBWBWBWB BWBWBWBW WBWBWBWB
output
YES
input
WBWBWBWB WBWBWBWB BBBWWWWB BWBWBWBW BWBWBWBW BWBWBWWW BWBWBWBW BWBWBWBW
output

### → Attention

Package for this problem was not updated by the problem writer or Codeforces administration after we've upgraded the judging servers. To adjust the time limit constraint, solution execution time will be multiplied by 2. For example, if your solution works for 400 ms on judging servers, then value 800 ms will be displayed and used to determine the verdict.

### Codeforces Round #157 (Div. 2)

Finished

### → Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.



Start virtual contest

### → Problem tags

brute force strings

No tag edit access

### → Contest materials

- Announcement 
- Tutorial 

NO

**Note**

In the first sample you should shift the following lines one position to the right: the 3-rd, the 6-th, the 7-th and the 8-th.

In the second sample there is no way you can achieve the goal.