3663 Connect It, If You Can!

"Connect it" is a popular game for all ages. Newman loves the game very much. "Connect it" has a stronger version with slightly different rules.

The game venue is expressed by size. When the game starts, there are $2size \times 2size$ diamonds, each diamond is a square whose side length is 1 with different patterns on it. The diamonds form a Big Square whose side length is $2size$. Each time, a player can choose two diamonds with the same patterns. If exists a broken line which can connect the center point of two diamonds with one turning at most and which doesn't go through any other diamond, these two diamonds will be eliminated at the same time. Such a broken line is called "valid broken line". Players continue to eliminate diamonds until all diamonds are eliminated or no diamonds can be eliminated.

To enhance visual effect, we set a boundary for game venue. The boundary is a circle whose center is the center of the Big Square and whose radius is $2size$. Any part of the broken line cannot go beyond the circle, but can be in (or on) the circle.

Newman's task is to complete a part of the game, i.e. for a given middle status or start status in the game, determine whether two diamonds can be connected by a "valid broken line". Suppose the two diamonds have the same patterns, Newman hopes you can help him.

Here are two examples of valid broken line:

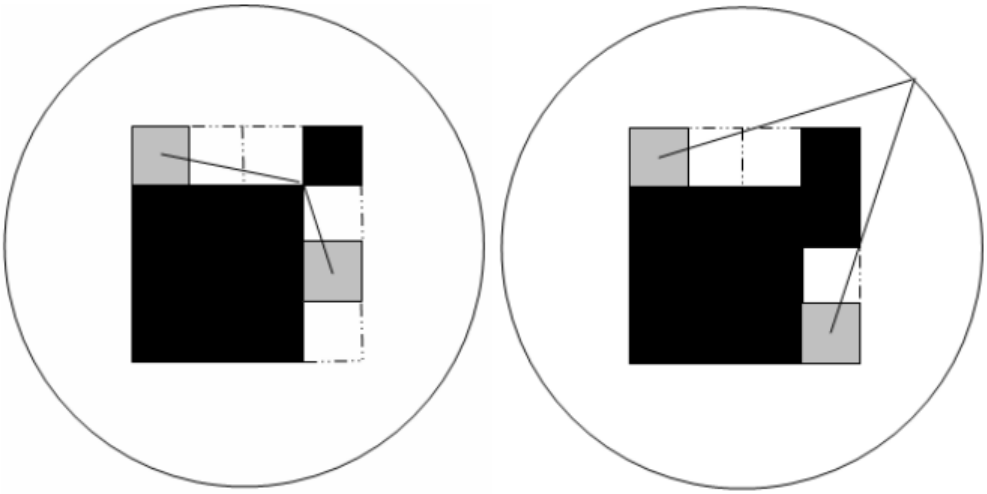

Figure 1: Two examples of valid broken line.

Input

Input contains several cases.

For each case, the first line is an integer $size$, $(1 \leq size \leq 50)$ to indicate the size of game venue.

A character matrix with $2size$ rows and $2size$ columns follows to indicate the $2size \times 2size$ places respectively.

'`.`' means the place is empty, otherwise there is a diamond. A normal diamond is indicated by '`#`'. The two diamonds to be connected are indicated by '`S`' and '`E`'. We assure there exactly one '`S`' and '`E`'. The total number of diamonds is no more than 400.

The last case is followed by a line containing only a zero.

Output

For each case, output '`Yes`' (quotes for clarity) if the valid broken line between the two diamonds exists or output '`No`' (quotes for clarity) otherwise. Please output in the following format.

Sample Input

```
2
S..#
###.
###E
###.
2
S..#
####
###E
###.
0
```

Sample Output

```
Case 1: Yes
Case 2: No
```