

Problem 7. The LOL Game (Small)

Time limit: 5 seconds
Memory limit: 256 megabytes

The LOL game is played as follows. Start with a blank piece of paper with a row of n empty boxes on it. The players alternate filling in a blank box with a letter L or a letter O. Once a box is filled, it remains the same and can never be used again. The goal is to be the first player to form a pattern LOL among three consecutive boxes.

Assuming both players play with an optimal strategy for themselves, the result of the game is defined, and can be a win for the next player to play, a draw (nobody wins), or a win for the previous player. We denote these three possibilities with the letters N for the Next player having a winning strategy, D for a Draw, and P for Previous player having a winning strategy. We say the value of a position is N , D , or P as defined here.

For example, suppose the state of the game is LL-L. (The “-” symbol indicates an empty box.) Then the next player puts a O in the blank and wins the game. Therefore LL-L is an N position. On the other hand, consider L--L. Now no matter what the next player does, the other player wins, so L--L is a P position.

In this problem you are to write a program that can compute the value of any position that might occur in an LOL game of limited size.

Input

The first line of input is n , the number of boxes in the game. The next line is a sequence of n characters from the set {L, O, -}. A “-” indicates an empty box. Here n is at most 26 and the number of “-” characters is at most 7. It is guaranteed that the position is not a terminal one. In other words it will not contain the winning LOL pattern, and it will contain at least one “-”.

Output

The output is a line containing N, D, or P, the value of the game.

Examples

standard input	standard output
4 LL-L	N
4 L--L	P
7 -----	N

Note

In the last example, a winning move for the next player is to put an L in the middle box.