

## A. Between the Offices

time limit per test: 2 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

As you may know, MemSQL has American offices in both San Francisco and Seattle. Being a manager in the company, you travel a lot between the two cities, always by plane.

You prefer flying from Seattle to San Francisco than in the other direction, because it's warmer in San Francisco. You are so busy that you don't remember the number of flights you have made in either direction. However, for each of the last  $n$  days you know whether you were in San Francisco office or in Seattle office. You always fly at nights, so you never were at both offices on the same day. Given this information, determine if you flew more times from Seattle to San Francisco during the last  $n$  days, or not.

### Input

The first line of input contains single integer  $n$  ( $2 \leq n \leq 100$ ) — the number of days.

The second line contains a string of length  $n$  consisting of only capital 'S' and 'F' letters. If the  $i$ -th letter is 'S', then you were in Seattle office on that day. Otherwise you were in San Francisco. The days are given in chronological order, i.e. today is the last day in this sequence.

### Output

Print "YES" if you flew more times from Seattle to San Francisco, and "NO" otherwise.

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

### Examples

<b>input</b>
4 FSSF
<b>output</b>
NO

  

<b>input</b>
2 SF
<b>output</b>
YES

  

<b>input</b>
10 FFFFFFFFFF
<b>output</b>
NO

  

<b>input</b>
10 SSFFSFFSFF
<b>output</b>
YES

## Note

In the first example you were initially at San Francisco, then flew to Seattle, were there for two days and returned to San Francisco. You made one flight in each direction, so the answer is "NO".

In the second example you just flew from Seattle to San Francisco, so the answer is "YES".

In the third example you stayed the whole period in San Francisco, so the answer is "NO".

In the fourth example if you replace 'S' with ones, and 'F' with zeros, you'll get the first few digits of  $\pi$  in binary representation. Not very useful information though.