Problem 1: Robot Routing

A robot is sitting in the upper left hand corner of an $N \times N$ grid. The robot can move in only two directions: right and down.

However, there are obstacles occupying several of the grid squares that prevent the robot from using that square. How many possible paths can the robot take to the lower right corner?

Input
The first line contains the integers N, M ($1 \leq N \leq 1000, 1 \leq M \leq 1000$). N is the size of the grid, and M is the number of obstacles in the grid.

The next M lines each contain two integers, i, j, the location of a new obstacle.

Output
Print the total number of paths from the upper left to the lower right, modulo $10^9 + 7$.

Problem 2: Password Palindromes

Anna loves palindromes. (A palindrome is a string that reads the same forwards and backwards.) Every time she makes a new password, she uses a new palindrome. Then she writes it down next to her computer.

However, for privacy reasons, when she writes it down, she also adds random characters at various locations, so that no one will guess her password.

Unfortunately, Anna has forgotten one of her passwords. Help her recover it from her notes.

Input
The first line contains an integer n ($1 \leq n \leq 1000$), the length of a string. The next line contains the string.

Output
Print the longest possible palindrome that is contained in the characters of the string. If there are multiple palindromes that are equally long, print any of them.

Problem 3: Password Palindromes, Part 2

Anna is grateful for your help! But now she has a large pile of notes with different passwords next to her computer. (She knows, of course, that re-using passwords is very foolish.)

She wants to combine all of her notes into a single string that contains all of her passwords. But then she wonders, how will she recover all of the possible passwords from her notes? Help her decide whether this is feasible. For simplicity, she will only include the passwords themselves, and not add random characters this time.

Input
The first line contains an integer n ($1 \leq n \leq 1000$), the length of the string. The next line contains the string with all of her palindromic passwords concatenated together.

Output
Count the number of possible ways to break up the input string into palindromes. Note: Strings of length 1 are still considered palindromes.