Sword Game

It is 2048, and a Martian robot called SSH3 wants to win galactical wars, as he does every year.
He has been chosen to participate in the prestigious Sword Game, possibly the last one to ever happen. This is the format of the Sword Game.
Every sword has a name S which is a string of n characters from a-z. The strength of the sword is decided in the following way.
Define a function f (l1, l2) for every 1<= l1, l2 <= n. Find all the substrings of the sword name S with length l1. Then, ind1 is the index such that it is the lexicographically smallest among these l1 length substrings. (If there are multiple such substrings, then we consider the **lower** index).
Similarly, ind2 is defined. Then, f (l1, l2) = |ind1 - ind2|. (All indices are 0-based)

Strength (S) = (Expected Value (f)), over all l1, l2.

As a Martian, SSH3 is also expected to be very good at Maths. He is asked to find out the strength of the sword. But as all other Martians, he has come from the city of Quoda, where everybody is pathetic at Maths. Hence, he asks you to help him. Ouput the value of n^2 * Strength (S).

Input:
First line contains the integer n.
Second line contains the name of the sword S.

Output:
A single line containing an **Integer**, that is the value of n^2 * Strength (S).

Constraints:
1 <= n <= 100000
S contains exactly n characters from ‘a’ to ‘z’.

Time Limit: 8 seconds.

Examples:

Input:
2
ab

Output:
0

Explanation: f(l1,l2) can take one of the following 4 values:
f(1,1) = 0 (ind1 = 0, ind2 = 0)
f(1,2) = 0 (ind1 = 0, ind2 = 0)
f(2,1) = 0 (ind1 = 0, ind2 = 0)
f(2,2) = 0 (ind1 = 0, ind2 = 0)
E(f(l1,l2)) = 0
2^2 * E(f(l1,l2)) = 0