

```

1 //
2 // Let N be a positive integer. You are given a string s of length N-1, consisting of < and >.
3 // Find the number of permutations (p[1], p[2], ... p[N]) of (1,2, ... N)
4 // that satisfy the following condition, modulo 10^9 + 7
5 // For each i (1 ≤ i ≤ N-1), p[i] < p[i+1] if the i-th character in s is '<'
6 // and p[i] > p[i+1] if the i-th character in s is '>'
7 //
8 // Time Complexity: O(N^2)
9 //
10
11 #include <bits/stdc++.h>
12 #define ll long long
13 using namespace std;
14 const int MOD = 1e9 + 7;
15
16 inline void uadd(int& a, int b) {
17     a = (a + b);
18     if (a ≥ MOD) a -= MOD;
19 }
20
21 int main() {
22     int n;
23     cin >> n;
24
25     string s;
26     cin >> s;
27     s = " " + s;
28
29     vector<vector<int>> dp(n, vector<int>(n, 0));
30     dp[0][0] = 1;
31
32     // dp[i][j] - number of ways to order the first i elements ending with item j
33     for (int i = 1; i < n; i++) {
34         int pref = 0;
35         for (int j = 0; j ≤ i; j++) { // last number
36             if (s[i] == '<') {
37                 uadd(dp[i][j], pref);
38                 uadd(pref, dp[i-1][j]);
39             } else {
40                 uadd(dp[i][i-j], pref);
41                 uadd(pref, dp[i-1][i-j-1]);
42             }
43         }
44     }
45
46     int ans = 0;
47     for (int i = 0; i < n; i++) {
48         uadd(ans, dp[n-1][i]);
49     }
50
51     cout << ans << endl;
52     return 0;
53 }

```