Problem 3320: Count The Number of Winning Sequences

Problem Information

Difficulty: Hard
Acceptance Rate: 31.76%
Paid Only: No
Tags: String, Dynamic Programming

Problem Description

Alice and Bob are playing a fantasy battle game consisting of `n` rounds where they summon one of three magical creatures each round: a Fire Dragon, a Water Serpent, or an Earth Golem. In each round, players **simultaneously** summon their creature and are awarded points as follows:

* If one player summons a Fire Dragon and the other summons an Earth Golem, the player who summoned the **Fire Dragon** is awarded a point. * If one player summons a Water Serpent and the other summons a Fire Dragon, the player who summoned the **Water Serpent** is awarded a point. * If one player summons an Earth Golem and the other summons a Water Serpent, the player who summoned the **Earth Golem** is awarded a point. * If both players summon the same creature, no player is awarded a point.

You are given a string `s` consisting of `n` characters `'F'`, `'W'`, and `'E'`, representing the sequence of creatures Alice will summon in each round:

* If `s[i] == 'F'`, Alice summons a Fire Dragon. * If `s[i] == 'W'`, Alice summons a Water Serpent. * If `s[i] == 'E'`, Alice summons an Earth Golem.

Bob's sequence of moves is unknown, but it is guaranteed that Bob will never summon the same creature in two consecutive rounds. Bob _beats_ Alice if the total number of points awarded to Bob after `n` rounds is **strictly greater** than the points awarded to Alice.

Return the number of distinct sequences Bob can use to beat Alice.

Since the answer may be very large, return it **modulo** `109 + 7`.

Example 1:

Input: s = "FFF"

Output: 3

Explanation:

Bob can beat Alice by making one of the following sequences of moves: `"WFW"`, `"FWF"`, or `"WEW"`. Note that other winning sequences like `"WWE"` or `"EWW"` are invalid since Bob cannot make the same move twice in a row.

Example 2:

Input: s = "FWEFW"

Output: 18

Explanation:

Bob can beat Alice by making one of the following sequences of moves: `"FWFWF"`, `"FWFWE"`, `"FWEFE"`, `"FWEWE"`, `"FEFWF"`, `"FEFWE"`, `"FEFEW"`, `"FEWFE"`, `"WFEFE"`, `"WFEWE"`, `"WEFWF"`, `"WEFWE"`, `"WEFEF"`, `"WEFEW"`, `"WEWFW"`, `"WEWFE"`, `"EWFWE"`, or `"EWEWE"`.

Constraints:

* `1 <= s.length <= 1000` * `s[i]` is one of `'F'`, `'W'`, or `'E'`.

Code Snippets

C++:

```
class Solution {
public:
int countWinningSequences(string s) {

}
};
```

Java:

```
class Solution {
public int countWinningSequences(String s) {

}
}
```

Python3:

```
class Solution:
def countWinningSequences(self, s: str) -> int:
```