Problem 3443: Maximum Manhattan Distance After K Changes

Problem Information

Difficulty: Medium
Acceptance Rate: 54.14%
Paid Only: No
Tags: Hash Table, Math, String, Counting

Problem Description

You are given a string `s` consisting of the characters `'N'`, `'S'`, `'E'`, and `'W'`, where `s[i]` indicates movements in an infinite grid:

* `'N'` : Move north by 1 unit. * `'S'` : Move south by 1 unit. * `'E'` : Move east by 1 unit. * `'W'` : Move west by 1 unit.

Initially, you are at the origin `(0, 0)`. You can change **at most** `k` characters to any of the four directions.

Find the **maximum** **Manhattan distance** from the origin that can be achieved **at any time** while performing the movements **in order**.

The **Manhattan Distance** between two cells `(xi, yi)` and `(xj, yj)` is `|xi - xj| + |yi - yj|`.

Example 1:

Input: s = "NWSE", k = 1

Output: 3

Explanation:

Change `s[2]` from `'S'` to `'N'`. The string `s` becomes `"NWNE"`.

Movement | Position (x, y) | Manhattan Distance | Maximum ---|---|---|--- s[0] == 'N' | (0, 1) | 0 + 1 = 1 | 1 s[1] == 'W' | (-1, 1) | 1 + 1 = 2 | 2 s[2] == 'N' | (-1, 2) | 1 + 2 = 3 | 3 s[3] == 'E' | (0, 2) | 0 + 2 = 2 | 3 The maximum Manhattan distance from the origin that can be achieved is 3. Hence, 3 is the output.

Example 2:

Input: s = "NSWWEW", k = 3

Output: 6

Explanation:

Change `s[1]` from `'S'` to `'N'`, and `s[4]` from `'E'` to `'W'`. The string `s` becomes `"NNWWWW"`.

The maximum Manhattan distance from the origin that can be achieved is 6. Hence, 6 is the output.

Constraints:

* `1 <= s.length <= 105` * `0 <= k <= s.length` * `s` consists of only `'N'`, `'S'`, `'E'`, and `'W'`.

Code Snippets

C++:

```
class Solution {
public:
int maxDistance(string s, int k) {

}
};
```

Java:

```
class Solution {
public int maxDistance(String s, int k) {

}
```

```
}
```

Python3:

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