

# 3443. Maximum Manhattan Distance After K Changes

Solved

Medium Topics Hint

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

Python3 Auto

```
1 class Solution:
2     def maxDistance(self, s: str, k: int) -> int:
3
4         ''' the manhattan distacne is dynamically recorded
5         c = Counter(s)
6         #print(c)
7         h_min = min(c['E'], c['W'])
8         v_min = min(c['N'], c['S'])
9
10
11         return abs(c['E'] - c['W']) + abs(c['N'] - c['S']) + 2 * min(k, h_min + v_min)
12         '''
13
14         h, v = 0, 0
15         res = 0
16
17         for i in range(len(s)):
18             if s[i] == 'N':
19                 v += 1
20             elif s[i] == 'S':
21                 v -= 1
22             elif s[i] == 'E':
23                 h += 1
24             elif s[i] == 'W':
25                 h -= 1
26
27         res = max(res, min(abs(v)+abs(h)+2*k, i+1))
28
29         return res
30
```

