

Problem 3119: Maximum Number of Potholes That Can Be Fixed

Problem Information

Difficulty: Medium

Acceptance Rate: 53.33%

Paid Only: Yes

Tags: String, Greedy, Sorting

Problem Description

You are given a string `road`, consisting only of characters `"x"` and `"."`, where each `"x"` denotes a _pothole_ and each `"."` denotes a smooth road, and an integer `budget`.

In one repair operation, you can repair `n` **consecutive** potholes for a price of `n + 1`.

Return the **maximum** number of potholes that can be fixed such that the sum of the prices of all of the fixes **doesn't go over** the given budget.

Example 1:

Input: road = "..", budget = 5

Output: 0

Explanation:

There are no potholes to be fixed.

Example 2:

Input: road = "..xxxxx", budget = 4

Output: 3

****Explanation:****

We fix the first three potholes (they are consecutive). The budget needed for this task is ` $3 + 1 = 4$ `.

****Example 3:****

****Input:**** road = "x.x.xxx...x", budget = 14

****Output:**** 6

****Explanation:****

We can fix all the potholes. The total cost would be `(1 + 1) + (1 + 1) + (3 + 1) + (1 + 1) = 10` which is within our budget of 14.

****Constraints:****

* `1 <= road.length <= 105` * `1 <= budget <= 105 + 1` * `road` consists only of characters `'.` and `'x'`.

Code Snippets

C++:

```
class Solution {
public:
    int maxPotholes(string road, int budget) {
        }
};
```

Java:

```
class Solution {
public int maxPotholes(String road, int budget) {
    }
}
```

Python3:

```
class Solution:  
    def maxPotholes(self, road: str, budget: int) -> int:
```