1208. Get Equal Substrings Within Budget

Description

You are given two strings `s` and `t` of the same length and an integer `maxCost`.

You want to change `s` to `t`. Changing the `i th` character of `s` to `i th` character of `t` costs `|s[i] - t[i]|` (i.e., the absolute difference between the ASCII values of the characters).

Return *the maximum length of a substring of* `s` *that can be changed to be the same as the corresponding substring of* `t` *with a cost less than or equal to* `maxCost`. If there is no substring from `s` that can be changed to its corresponding substring from `t`, return `0`.

Example 1:

```
Input: s = "abcd", t = "bcdf", maxCost = 3
Output: 3
Explanation: "abc" of s can change to "bcd".
That costs 3, so the maximum length is 3.
```

Example 2:

```
Input: s = "abcd", t = "cdef", maxCost = 3
Output: 1
Explanation: Each character in s costs 2 to change to character in t,  so the maximum length is 1.
```

Example 3:

```
Input: s = "abcd", t = "acde", maxCost = 0
Output: 1
Explanation: You cannot make any change, so the maximum length is 1.
```

Constraints:

- `1 <= s.length <= 10 5`
- `t.length == s.length`
- `0 <= maxCost <= 10 6`
- `s` and `t` consist of only lowercase English letters.