Problem 1871: Jump Game VII

Problem Information

Difficulty: Medium
Acceptance Rate: 26.06%
Paid Only: No
Tags: String, Dynamic Programming, Sliding Window, Prefix Sum

Problem Description

You are given a **0-indexed** binary string `s` and two integers `minJump` and `maxJump`. In the beginning, you are standing at index `0`, which is equal to `'0'`. You can move from index `i` to index `j` if the following conditions are fulfilled:

* `i + minJump <= j <= min(i + maxJump, s.length - 1)`, and * `s[j] == '0'`.

Return `true` _if you can reach index_`s.length - 1` _in_`s` _, or_`false` _otherwise._

Example 1:

Input: s = "_0_ 11 _0_ 1 _0_ ", minJump = 2, maxJump = 3 **Output:** true **Explanation:** In the first step, move from index 0 to index 3. In the second step, move from index 3 to index 5.

Example 2:

Input: s = "01101110", minJump = 2, maxJump = 3 **Output:** false

Constraints:

* `2 <= s.length <= 105` * `s[i]` is either `'0'` or `'1'`. * `s[0] == '0'` * `1 <= minJump <= maxJump < s.length`

Code Snippets

C++:

```
class Solution {
public:
bool canReach(string s, int minJump, int maxJump) {

}
};
```

Java:

```
class Solution {
public boolean canReach(String s, int minJump, int maxJump) {

}
}
```

Python3:

```
class Solution:
def canReach(self, s: str, minJump: int, maxJump: int) -> bool:
```