**Time Complexity of Z-Algorithm**

The time complexity of the Z algorithm for pattern searching is typically **O(m + n) ,** where m is the length of the pattern and n is the length of the text. This linear time complexity makes the Z algorithm efficient for pattern searching in many practical scenarios.

Let's break down the time complexity of the Z algorithm:

* Constructing the Z Array:

The main time-consuming step of the Z algorithm is constructing the Z array, which is done in O(n) time, where n is the length of the concatenated string (pattern + text). This is because we iterate over each character of the concatenated string once to compute the Z values.

* Pattern Matching:

After constructing the Z array, the time complexity for pattern matching is O(n), where n is the length of the concatenated string. This is because we iterate over the Z array once to identify occurrences of the pattern.

The time complexity of the Z algorithm is dominated by the construction of the Z array, which is O(n), where n is the length of the concatenated string. Once the Z array is constructed, pattern matching can be performed in linear time O(n). Therefore, the overall time complexity of the Z algorithm for pattern searching is O(m+n), where

m is the length of the pattern and n is the length of the text.

**Best Case Scenario:**

The best-case scenario for the Z-algorithm occurs when the pattern p does not occur or occurs rarely in the text s. In this case, the algorithm performs a linear scan through both the text and the pattern strings, calculating the Z-values without significant additional operations.

Time Complexity: O(n + m)

**Worst Case Scenario:**

The worst-case scenario for the Z-algorithm occurs when the pattern p matches the beginning of the text s at every position. In this case, the algorithm needs to calculate the Z-values for every position in the text, resulting in more computational overhead.

Time Complexity: O(n + m)

Overall, the Z-algorithm's time complexity remains linear in both the best and worst cases, making it an efficient choice for string searching tasks.