RABIN KARP ALGORITHM

Analysis of Time Complexity

- **Best and Average Case:** O(n + m) - This occurs when the hash function used by the algorithm minimizes collisions. In this case, the time to calculate the rolling hash for the text window and compare it to the pattern's hash dominates, taking O(n) time. Additionally, verification of potential matches using character-by-character comparison takes O(m) time.

- **Worst Case:** O(nm) - This arises when the hash function produces a high number of collisions. In this scenario, the algorithm might find a potential match (due to a hash collision) at almost every position in the text. It then needs to perform the character-by-character verification (O(m)) for each potential match, leading to a worst-case complexity of O(nm). This typically happens with poorly designed hash functions or specific patterns in the text that cause frequent collisions.

Here's a breakdown of the factors affecting time complexity:

n: Length of the text

m: Length of the pattern