The program performs **motif discovery** on textual or biological datasets.
Its goal is to find repeating patterns (motifs) in a long sequence using two different similarity concepts:

1. **Parameterized Matching** – detects substrings that share the same *structure* of repeated characters (the relative order of appearance is preserved, even if the actual characters are different).
2. **Abelian Matching** – detects substrings that contain the same *set and frequency* of characters as the pattern, regardless of their order.

The program:

Parameterized Matching

This algorithm detects substrings that share the same internal structure of repeated characters. While the actual characters may differ, their relative order and the positions of repetitions must be preserved.

Abelian Matching

This algorithm identifies substrings that contain the same multiset and frequency of characters as the pattern. It ignores the specific order of the characters, effectively finding all permutations or anagrams of the pattern.

Comparative Results

For each dataset, the program identifies all matching positions for both methods and determines the "Common Motif Positions" where both criteria are met.

Conclusions:

If we look at your results for test1.txt (the highly repetitive sequence):

- **Abelian Matching:** Found **325**matches for the pattern.
- **Parameterized Matching:** Found **96** matches for the motif.

While Abelian matching shows more matches numerically, this happens because the test data was designed for exact matching. This specific dataset contains many repeating sequences, which creates an environment that heavily favors permutations (anagrams).

- Parameterized Matching is more flexible: It is alphabet-independent. It can detect a much higher volume of motifs because it allows characters to be "renamed".
- Abelian Matching is stricter: It is composition-dependent. It requires an exact match of character identities and frequencies. While it is flexible regarding the *order* of characters (anagrams), it is restricted to the specific letters provided in the pattern

Example of result:

Dataset 1 (test1.txt) example:

- **Abelian Matching:**
  - Pattern: ACGTA
  - Found: ATGCA at Position 1.
  - ATGCA is a anagram of ACGTA (Counts: A:2, C:1, G:1, T:1).
- **Parameterized Matching:**
  - Motif Identified: ATGCA (96 occurrences).
  - Structural Insight: Parameterized matching successfully matched ATGCA with TGCAT. Even though they are not anagrams, they share the same Structure Signature (0, 0, 0, 0, 4). This demonstrates that Parameterized matching is focused on the "shape" of the string (where characters repeat) rather than the characters themselves.

Due to the extensive volume of data generated, the console output in the command prompt is truncated. To review the complete set of results, please refer to the output.txt file