Summary on methods, algorithms, and data structures: For assignment 5

ASSIGNMENT NUMBER : 05
NAME : Dinesh Raj Pampati
ULID : dpampat
NAME OF SECRET DIRECTORY : namo

Summary on methods, algorithms, and data structures:

1. **Methods Used**:
    - **main(String[] args)**: The driver method, reads RNA sequences from a file, calculates the optimal foldings using dynamic programming, and prints the computed matrix for each RNA sequence.
2. **Algorithm**:
    - **RNA Folding Algorithm**:
        1. Initialize a 2D matrix with dimensions equivalent to the length of the RNA sequence.
        2. For diagonal elements, initialize with 0 as they represent a single nucleotide.
        3. Loop through the RNA sequence with a difference 'i' between nucleotides.
        4. For each pair of nucleotides **(j, k)**:
            - Check for valid pairing, ensuring difference **k-j** is greater than or equal to 5.
            - Update the matrix value based on adjacent values and possible foldings.
        5. Use the computed matrix to determine the maximum number of possible foldings for the RNA sequence.
        6. From the about output matrix I will find the base pairs
3. **Data Structures Used**:
    - **String**: To store the RNA sequence.
    - **2D int array (int[][])**: Used to represent the matrix that holds the number of possible foldings between nucleotides.
    - **File**: Used to specify the file path for the input.
    - **Scanner**: Used to read input from the file.
4. **Time Complexity**:
    - **RNA Folding Algorithm**:
        - Initialization of matrix: *O*(*n*), where *n* is the length of the RNA sequence.
        - Main Loop (dynamic programming loop): *O*(*n*3) since there are 3 nested loops for traversing and updating the matrix.
        - Overall complexity for RNA Folding: *O*(*n*+*n*3)≈*O*(*n*3).
5. **Space Complexity**:

- **RNA Folding Algorithm**:
    - 2D Matrix: $O(n2)$, where n is the length of the RNA sequence.
    - RNA Sequence String: $O(n)$.
    - Overall space complexity: $O(n+n2)\approx O(n2)$.