Valid Parenthesis LeetCode

This program demonstrates two approaches to check if a given expression containing parentheses (round brackets **()**, curly brackets **{}**, and square brackets **[]**) is valid or not. A valid expression has matching pairs of opening and closing brackets.

Approach 1: Function to check balanced parentheses using a stack-based approach

In this approach, a stack is used to keep track of opening brackets as they are encountered. Whenever a closing bracket is encountered, it is checked whether it matches the top element of the stack, which should be the corresponding opening bracket. If they match, the opening bracket is popped from the stack. If they don't match, the expression is invalid. If, at the end, the stack is empty, the expression is valid.

Time Complexity:

- **The loop iterates through each character in the expression once, resulting in a linear time complexity of O(n), where n is the length of the expression.**

Space Complexity:

- **The stack is used to store opening brackets, and in the worst case, the stack could hold all opening brackets from the expression, resulting in a space complexity of O(n), where n is the length of the expression.**

Approach 2: Function to check balanced parentheses using a hash map-based approach

In this approach, a hash map is used to define the mapping between opening and closing brackets. As characters are processed, opening brackets are pushed onto the stack. When a closing bracket is encountered, it is checked against the corresponding bracket defined in the hash map. If they match, the opening bracket is popped from the stack. If they don't match, the expression is invalid. If, at the end, the stack is empty, the expression is valid.

Time Complexity:

- Similar to the stack-based approach, **the loop iterates through each character in the expression once, resulting in a linear time complexity of O(n)**, where n is the length of the expression.

Space Complexity:

- In this approach, **the space complexity is also O(n) due to the potential worst-case scenario where all opening brackets are pushed onto the stack.**