

Stack Problem Statement

1. Matching Parentheses: One classic use case for a stack is to check whether parentheses in a given string are balanced or not. This problem is solved using a stack where each opening bracket is pushed into the stack and each closing bracket is popped out of the stack. If the stack is empty at the end of the string, then the parentheses are balanced. If there are any remaining brackets in the stack, then the parentheses are not balanced.

Input: " $(2 + 3) * 5$ "

Output: Balanced

Input: " $(2 + 3) * 5))$ "

Output: Not balanced

Input: "{[()()]}"

Output: Balanced

Input: "{[()]}"

Output: Not balanced

Input: "{[()[]{}]()}"

Output: Balanced

Input: "}"

Output: Not balanced