<EECS 348 Term Project>

<Arithmetic Expression Evaluation>

Test Case

Version <1.2>

Revision History

Date	Version	Description	Author
<01/DEC/2023>	<1.0>	<First version uploaded, decided test cases to use>	<Zach Sevart, Lingfeng Li, Asa Maker>
<02/DEC/2023>	<1.1>	<Update the output table>	<Zach Sevart, Lingfeng Li, Asa Maker>
<03/DEC/2023>	<1.2>	<Final review of the document before submission>	<Zach Sevart, Lingfeng Li, Asa Maker>

Table of Contents

Test Case

1. Purpose

The purpose of this Test Case Specification is to outline specific test cases for the Arithmetic Expression Evaluator project. This document aims to ensure that the program correctly parses and evaluates various arithmetic expressions, handles errors gracefully, and meets all specified requirements.

2. Test case identifier

Test Case ID	Expression	Description
TC01	3 + 4	Basic addition
TC02	8 - (5 - 2)	Subtraction with parentheses
TC03	10 * 2 / 5	Mixed multiplication and division
TC04	2 ^ 3	Exponentiation
TC05	4 * (3 + 2) % 7 - 1	Complex expression with mixed operators
TC06	(((2 + 3))) + (((1 + 2)))	Addition with extraneous parentheses
TC07	((5 * 2) - ((3 / 1) + ((4 % 3))))	Complex expression with mixed operators
TC08	(((2 ^ (1 + 1)) + ((3 - 1) ^ 2)) / ((4 / 2) % 3))	Nested parentheses with exponents
TC09	(((((5 - 3))) * (((2 + 1))) + ((2 * 3))))	Complex expression with mixed operators
TC10	((9 + 6)) / ((3 * 1) / (((2 + 2))) - 1)	Complex division expression
TC11	+(-2) * (-3) - ((-4) / (+5))	Expression with unary operators
TC12	-(+1) + (+2)	Expression with unary negation
TC13	-(-(-3)) + (-4) + (+5)	Complex expression with unary negation
TC14	+2 ^ (-3)	Exponentiation with unary operator
TC15	-(+2) * (+3) - (-4) / (-5)	Complex expression with unary operators
TC16	2 * (4 + 3 - 1	Error case: Unmatched parentheses
TC17	* 5 + 2	Error case: Insufficient operands
TC18	4 / 0	Error case: Division by zero

TC19	5 (2 + 3)	Error case: Missing operator
TC20	7 & 3	Error case: Invalid character
TC21	(((3 + 4) - 2) + (1	Error case: Unmatched parentheses
TC22	((5 + 2) / (3 * 0))	Error case: Division by zero
TC23	((2 -) 1 + 3)	Error case: Missing operand
TC24	((4 * 2) + (-))	Error case: Missing operand after unary operator
TC25	((7 * 3) @ 2)	Error case: Invalid character

3. Test item

Item: Arithmetic Expression Evaluator Features to be Tested:

- **Parsing and Calculating Arithmetic Expressions:**

a) Requirements Specification: The system should be able to parse arithmetic expressions containing operators (+ - * / % ^) and numeric constants, taking into account operator precedence and parentheses. (Refer to "Expression Parsing" in the Software Requirements Specifications document).

b) Design Specification: Overview of the system design, focusing on components like the parser and calculator. (Refer to "Overall System Design" and "Components and Structure" in the Software Architecture document)

c) User's Guide: Instructions on inputting arithmetic expressions for evaluation. (Refer to User Guide Section 3.1)

- **Robustness in Error Scenarios:**

a) Requirements Specification: The system should identify and handle invalid arithmetic expressions, providing appropriate error messages. (Refer to "Error Handling" in the Software Requirements Specifications document) and (Refer to "Subset 5: Error Handling" in the SRS document).

b) Design Specification: Describes system behavior, including error detection during operation. (Refer to "How it Works in Action" in the Software Architecture document)
c) User's Guide: Information on error messages and troubleshooting. (Refer to User Guide Section 4.1)

4. Input specifications

Inputs:

- **Values: Various** types of arithmetic expressions as indicated in test cases TC01 - TC23.
- **Related Inputs:** All inputs are independent strings representing arithmetic expressions.

- **Relationships Between Inputs:** Each input string represents a separate arithmetic expression, with no dependencies between inputs.

5. Output specifications

Test Case ID	Expression	Expected Result	Actual Result	Pass/fail
TC01	3 + 4	7	7	Pass
TC02	8 - (5 - 2)	5	5	Pass
TC03	10 * 2 / 5	4	4	Pass
TC04	2 ^ 3	8	8	Pass
TC05	4 * (3 + 2) % 7 - 1	5	5	Pass
TC06	(((2 + 3))) + (((1 + 2)))	8	8	Pass
TC07	((5 * 2) - ((3 / 1) + ((4 % 3))))	6	6	Pass
TC08	(((2 ^ (1 + 1)) + ((3 - 1) ^ 2)) / ((4 / 2) % 3))	4	4	Pass
TC09	(((((5 - 3))) * (((2 + 1))) + ((2 * 3))))	12	12	Pass
TC10	((9 + 6)) / ((3 * 1) / (((2 + 2))) - 1)	-60	-60	Pass
TC11	+(-2) * (-3) - ((-4) / (+5))	6.8	6.8	Pass
TC12	-(+1) + (+2)	1	1	Pass
TC13	-(-(-3)) + (-4) + (+5)	-2	-2	Pass
TC14	+2 ^ (-3)	0.125	0.125	Pass
TC15	-(+2) * (+3) - (-4) / (-5)	-6.8	-6.8	Pass
TC16	2 * (4 + 3 - 1	Mismatched parentheses detected.	Mismatched parentheses detected.	Fail
TC17	* 5 + 2	Insufficient operands for binary operator: *	Insufficient operands for binary operator: *	Fail
TC18	4 / 0	Division by zero.	Division by zero.	Fail
TC19	5 (2 + 3)	Error in evaluation. Stack has unexpected size: 2	Error in evaluation. Stack has unexpected size: 2	Fail
TC20	7 & 3	Invalid character detected: &	Invalid character detected: &	Fail

TC21	(((3 + 4) - 2) + (1	Mismatched parentheses detected.	Mismatched parentheses detected.	Fail
TC22	((5 + 2) / (3 * 0))	Division by zero.	Division by zero.	Fail
TC23	((2 -) 1 + 3)	Error in evaluation. Stack has unexpected size: 2	Error in evaluation. Stack has unexpected size: 2	Fail
TC24	((4 * 2) + (-))	Missing operand after unary operator: -	Missing operand after unary operator: -	Fail
TC25	((7 * 3) @ 2)	Invalid character detected: @	Invalid character detected: @	Fail