EECS 348 Group 16

Arithmetic Expression Calculator

Test Cases

Version 1.0

Revision History

Date	Version	Description	Author
3/12/2023	1.0	Document Creation	Mark Kitchin, Sam Muehlebach, Josh Welicky, Jennifer Aber, Jawad Ahsan, Basim Arshad

Table of Contents

Test Cases

1. Purpose

The purpose of this document is to show the results of a variety of test cases that were executed in the program based on the project requirements. All the test cases are defined by a test case ID, with each case containing a description, test data, expected results, actual results, and if the test case passed or failed. These test cases help to find defects within the code of the program and if a case fails, the code is edited in response to failed test cases.

2. Test case table

Test Case ID	Test Case Description	Test Data	Expected Results	Actual Results	Pass/Fail
STOP-1	Check halt mechanism	STOP	Ends Program	Ends Program	Pass
ADD-1	Check basic Addition	3 + 4	7	7	Pass
ADD-2	Check addition with unary negation	3 + (-4)	-1	-1	Pass
ADD-3	Complex Addition with Extraneous Parentheses	(((2 + 3))) + (((1 + 2)))	8	8	Pass
ADD-4	Check floating Addition	4.5 + 8.0	12.5	12.5	Pass
ADD-5	Check floating addition and integers	4.5 + 8	12.5	12.5	Pass
SUB-1	Check Basic Subtraction	4 - 3	1	1	Pass
SUB-2	Check subtraction with unary negation.	4 - (-2)	6	6	Pass
SUB-3	Subtraction with Parentheses	8 - (5 – 2)	5	5	Pass
SUB-4	Subtraction with floats	5.0 - 3.5	1.5	1.5	Pass
SUB-5	Subtraction with floats and integers	5 - 3.5	1.5	1.5	Pass
MULT-1	Check Basic Multiplication	2 * 2	4	4	Pass
MULT-2	Check multiplication with floats	2.0 * 5.0	10	10	Pass
MULT-3	Check multiplication with floats and integers	2 * 5.0	10	10	Pass
DIV-1	Check Basic Division	10 / 5	2	2	Pass

DIV-2	0 as Numerator	0 / 2	0	0	Pass
DIV-3	Division with floats	5 / 3.0	1.666667	1.666667	Pass
DIV-4	Division with floats and integers	5.0 / 3.0	1.666667	1.666667	Pass
COMB-1	Multiplication and Division	10 * 2 / 5	4	4	Pass
COMB-2	Mixed Operations	4 * (3 + 2) % 7 - 1	5	5	Pass
COMB-3	Mixed Operations with Extraneous parentheses	((5 * 2) - ((3 / 1) + ((4 % 3))))	6	6	Pass
COMB-4	Nested Parentheses with Exponents	(((2 ^ (1 + 1)) + ((3 - 1) ^ 2)) / ((4 / 2) % 3))	4	4	Pass
COMB-5	Combination of Extraneous and Necessary parentheses	(((((5 - 3))) * (((2 + 1))) + ((2 * 3))))	12	12	Pass
COMB-6	Extraneous Parentheses with Division	((9 + 6)) / ((3 * 1) / (((2 + 2))) - 1)	-60	-60	Pass
COMB-7	Combining Unary Operations with Arithmetic Operations	+(-2) * (-3) – ((-4) / (+5))	6.8	6.8	Pass
COMB-8	Unary Negation and Addition in Parentheses	-(+1) + (+2)	1	1	Pass
COMB-9	Negation and Addition with Negated Parentheses	-(-(-3)) + (-4) + (+5)	-2	-2	Pass
COMB-10	Unary Negation and Exponentiation	+2 ^ (-3)	0.125	0.125	Pass
COMB-11	Combining Unary Operations with Parentheses	-(+2) * (+3) - (-4) / (-5)	-6.8	-6.8	Pass
COMB-12	Checking multiple operations with floats	(5.0 * 2.0) ^ 2.0	100	100	Pass
EXPO-1	Check Basic Exponent	2 ^ 3	8	8	Pass

EXPO-2	Check Alternate Exponent Notation	2**3	8	8	Pass
EXPO-3	Detect excessive * in exponent.	2***3	Binary operator must be preceded by a number or closing parentheses.	A binary operator can only be preceded by a number or closing parentheses.	Pass
EXPO-4	Check exponentiation with floats	2.0 ^ 5.0	32	32	Pass
EXPO-5	Check exponentiation with floats and integers	2.0 ^ 5	32	32	Pass
EXPO-6	Check exponentiation with floats and integers	2 ^ 5.0	32	32	Pass
EXPO-7	Check exponentiation with a non-whole number	4 ^ 0.5	2	2	Pass
MODU-1	Check Basic Modulo	10 % 4	2	2	Pass
MODU-2	Check Modulo with 0 as output	10 % 10	0	0	Pass
MODU-3	Check Modulo with floating point whole numbers.	4.0%3.0	1	1	Pass
BREAK-1	Enter nothing	'\n'	No input detected	No Input detected	Pass
BREAK-2	Enter characters	F + G	Invalid Character detected	Invalid Character detected	Pass
BREAK-3	Using Modulo with decimal values	4.0 % 3.5	Modulo does not accept floats	Modulo does not accept floats	Pass
BREAK-4	Check Modulo 0	10 % 0	Cannot divide by zero	Program raised an exception	Pass

BREAK-5	Dividing by 0	5 / 0	Cannot divide by zero	Division by Zero is not allowed	Pass
BREAK-6	Unmatched Parentheses	2 * (4 + 3 - 2	Unmatched parentheses detected	Unmatched parentheses detected	Pass
BREAK-7	Operators without Operands	/ 5 + 2	A binary operator must be preceded by a number or closing parentheses.	A binary operator must be preceded by a number or closing parentheses	Pass
BREAK-8	Missing Operator	5 (2 + 3)	An opening parentheses can only be preceded by an operator or opening parentheses.	An opening parentheses can only be preceded by an operator or an opening parentheses.	Pass
BREAK-9	Invalid character	7 & 3	Invalid character detected.	Invalid character detected	Pass
BREAK-10	Mismatched Parentheses	(((3 + 4) - 2) + (1)	Unmatched parentheses detected.	Unmatched parentheses detected.	Pass
BREAK-11	Invalid Operator Usage	((5 + 2) / (3 * 0))	Zero division not allowed.	Division by zero is not allowed.	Pass
BREAK-12	Invalid Operator Sequence	((2 -) 1 + 3)	Closing parentheses must be preceded by a number or closing parentheses.	A closed parentheses can only be preceded by a number or closing parentheses.	Pass
BREAK-13	Missing Operand	((4 * 2) + (-))	Binary operator must be preceded by a number	A binary operator can only be preceded by a	Pass

			or closing parentheses.	number or closing parentheses.	
BREAK-14	Invalid Character	((7 * 3) @ 2)	Invalid character detected.	Invalid character detected	Pass
FLOAT-1	Prevent operands with multiple decimal points.	3.1.3 + 4	Multiple decimal points detected.	Multiple decimal points detected in one number.	Pass
FLOAT-2	Single decimal point	.	Single decimal detected	Invalid decimal point detected.	Pass
FLOAT-3	Check floats with no preceding number	.5+.5	1	1	Pass
FLOAT-4	Check no-preceding number floats with unary operator	-.5+1	0.5	0.5	Pass
UN-1	Check unguarded unary operator.	-3+-2	-5	-5	Pass
UN-2	Check properly placed unary operator	-4 + - 3	Binary operator must be preceded by a number or closing parentheses.	A binary operator can only be preceded by a number or closing parentheses.	Pass

3. Environmental needs

3.1.1 *Hardware*

N/A

3.1.2 *Software*

N/A

3.1.3 *Other*

There are no other specific environmental needs to run the test cases. As long as the device used to run these test cases has a command terminal, C++ compiler, and keyboard, all test cases can be run without any difficulty.

4. Special procedural requirements

N/A

5. Intercase dependencies

N/A