Arithmetic Expression Evaluator

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 29/11/2023 | 0.1 | The initial test cases | Alexandra, Deborah, Riley, Timo, Victor, Ellia |
| 03/12/2023 | 1.0 | The finished test case document | Alexandra, Deborah, Riley, Timo, Victor, Ellia |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Purpose 4

2. Test case identifier 4

3. Test item 4

4. Input specifications 4

5. Output specifications 4

6. Environmental needs 7

6.1.1 Hardware 7

6.1.2 Software 7

6.1.3 Other 7

7. Special procedural requirements 8

8. Intercase dependencies 8

# Purpose

This Test Case Specification document for the Arithmetic Expression Evaluator defines a test case for an item that should be tested.

This document will have eight sections: Purpose, Test case identifier, Test Item, Input specifications, Output specifications, Environmental needs, Special procedural requirements, Intercase dependencies. Sections two through four are condensed into the Output specifications section.

The output specifications section is a table with all of our test cases. It is organized as follows: the first column is the identifier for the test case; the second column is the description of the test case; the third column is the actual expression that will be plugged into the calculator; the fourth column is the value the expression should evaluate too; the fifth column is what the calculator returned; and the sixth column is whether the calculator failed or passed the test.

The remaining sections were not needed.

# Test case identifier

See the table.

# Test item

See the table.

# Input specifications

See the table.

# Output specifications

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test  Case ID | Test Case  Description | Test  Data | Expected  Results | Actual  Results | Pass/  Fail |
| Add-101 | Test unary operators inside parenthesis | (2+(-3)) | -1 | -1.000000 | Pass |
| Add-102 | Test unary operator outside parenthesis | -(2+(3)) | -5 | -5.000000 | Pass |
| Add-103 | Test unary operator inside and outside parenthesis | -(2+(-3)) | 1 | 1.000000 | Pass |
| Add-104 | Test addition with positive values | 6+7 | 13 | 13.0000 | Pass |
| Add-105 | Test addition with zero | 5+0 | 5 | 5.000000 | Pass |
| Add-106 | Test addition with decimals | 2.5+2.5 | 5 | 5.000000 | Pass |
| Sub-101 | Test unary operators inside parenthesis | (2-(-3)) | 5 | 5.000000 | Pass |
| Sub-102 | Test unary operator outside parenthesis | -(2-(3)) | 1 | 1.000000 | Pass |
| Sub-103 | Test unary operator inside and outside parenthesis | -(2-(-3)) | -5 | -5.000000 | Pass |
| Sub-104 | Test subtraction with positive values | 3-2 | 1 | 1.000000 | Pass |
| Sub-105 | Test subtraction with zero | 2-0 | 2 | 2.000000 | Pass |
| Sub-106 | Test subtraction with negative values | (-2)-(-3) | 1 | 1.000000 | Pass |
| Sub-107 | Test subtraction with decimal numbers | 5-2.5 | 2.5 | 2.500000 | Pass |
| Mult-101 | Test unary operators inside parenthesis | (2\*(-3)) | -6 | -6.000000 | Pass |
| Mult-102 | Test unary operator outside parenthesis | -(2\*(3)) | -6 | -6.000000 | Pass |
| Mult-103 | Test unary operator inside and outside parenthesis | -(2\*(-3)) | 6 | 6.000000 | Pass |
| Mult-104 | Test multiplication with positive values | 2\*3 | 6 | 6.000000 | Pass |
| Mult-105 | Test multiplication with zero | 2\*0 | 0 | 0.000000 | Pass |
| Mult-106 | Test multiplication with one | 2\*1 | 2 | 2.000000 | Pass |
| Mult-107 | Test multiplication with negative | 2\*(-3) | -6 | -6.000000 | Pass |
| Mult-108 | Test multiplication with decimal numbers | 2\*2.5 | 5 | 5.000000 | Pass |
| Mult-109 | Test multiplication with a negative number with no parentheses | 2\*-3 | -6 | -6.000000 | Pass |
| Div-101 | Test unary operators inside parenthesis | (9/(-3)) | -3 | -3.000000 | Pass |
| Div-102 | Test unary operator outside parenthesis | -(9/(3)) | -3 | -3.000000 | Pass |
| Div-103 | Test unary operator inside and outside parenthesis | -(9/(-3)) | 3 | 3.000000 | Pass |
| Div-104 | Test division by zero | 5/0 | “Division by zero is invalid.” | “Division by zero is invalid.” | Pass |
| Div-105 | Test infinite numbers | 1/3 | .33333 | 0.333333 | Pass |
| Div-106 | Test division by decimal numbers | 5/2.5 | 2 | 2.000000 | Pass |
| Mod-101 | Test unary operators inside parenthesis | (6%(-4)) | -2 | -2.000000 | Pass |
| Mod-102 | Test unary operator outside parenthesis | -(6%(4)) | -2 | -2.000000 | Pass |
| Mod-103 | Test unary operator inside and outside parenthesis | -(6%(-4)) | 2 | 2.000000 | Pass |
| Mod-104 | Test modulo by zero | 5%0 | “Modulus by zero is invalid.” | “Modulus by zero is invalid.” | Pass |
| Mod-105 | Test modulo with a negative denominator | 5%-2 | -1 | -1.000000 | Pass |
| Mod-106 | Test modulo with a negative numerator | -5%2 | 1 | 1.000000 | Pass |
| Mod-107 | Test modulo with both numbers as negative | -5%-2 | -1 | -1.000000 | Pass |
| Mod-108 | Test modulo with decimals | 5%2.3 | .4 | 0.400000 | Pass |
| Exp-101 | Test unary operators inside parenthesis | (2^(-3)) | .125 | 0.125000 | Pass |
| Exp-102 | Test unary operator outside parenthesis | -(2^(3)) | -8 | -8.000000 | Pass |
| Exp-103 | Test unary operator inside and outside parenthesis | -(2^(-3)) | -.125 | -0.125000 | Pass |
| Exp-104 | Testing exponent with a negative power | 2^-2 | .25 | 0.250000 | Pass |
| Exp-105 | Testing exponent with a positive power and base | 2^2 | 4 | 4.000000 | Pass |
| Exp-106 | Testing exponent with a negative base | -2^2 | 4 | 4.000000 | Pass |
| Exp-107 | Testing exponent with both a negative power and base | -2^-2 | .25 | 0.250000 | Pass |
| Exp-108 | Testing exponent with an equation in power | 2^(4+3\*1) | 128 | 128.000000 | Pass |
| Exp-109 | Testing exponent with a base that is a decimal | 0.5^2 | .25 | 0.250000 | Pass |
| Exp-109.1 | Testing exponent with a decimal with no number before the decimal point | .5^2 | .25 | 0.250000 | Pass |
| Exp-110 | Testing exponent with a power that is a decimal | 2^0.5 | 1.414214 | 1.414214 | Pass |
| Exp-110.1 | Testing exponent with a decimal with no number before the decimal point | 2^.5 | 1.414214 | 1.414214 | Pass |
| Exp-111 | Testing exponent with multiple in a row | 2^(2^3) | 256 | 256.000000 | Pass |
| Err-101 | Testing for an error that has unmatched opening and closing parentheses | 2\*(4+3-1 | “Unmatched parentheses.” | “Unmatched parentheses.” | Pass |
| Err-102 | Testing for operators without operands | \*5+2 | “Not enough operands.” | “Not enough operands.” | Pass |
| Err-103 | Testing for missing operators | 5(2+3) | “Not enough operators.” | “Not enough operators.” | Pass |
| Err-104 | Testing for invalid characters | 7&3 | “Invalid Character” | “Invalid Character” | Pass |
| Err-105 | Testing for mismatch parentheses | (((3+4)-2)+(1) | “Unmatched parentheses.” | “Unmatched parentheses.” | Pass |
| Err-106 | Testing for invalid operator usage | ((5+2)/(3\*0)) | “Division by zero is invalid.” | “Division by zero is invalid.” | Pass |
| Err-107 | Testing for invalid operator sequence | ((2-)1+3) | “Not enough operands.” | “Not enough operands.” | Pass |
| Err-108 | Testing for missing operands | ((4\*2)+(-)) | “Not enough operands.” | “Not enough operands.” | Pass |
| Err-109 | Testing for invalid characters | ((7\*3)@2) | “Invalid Character” | “Invalid Character” | Pass |
| Case-1 | Test addition | 3+4 | 7 | 7.000000 | Pass |
| Case-2 | Subtraction with Parenthesis | 8-(5-2) | 5 | 5.000000 | Pass |
| Case-3 | Multiplication and division | 10\*2/5 | 4 | 4.000000 | Pass |
| Case-4 | Exponentiation | 2^3 | 8 | 8.000000 | Pass |
| Case-5 | Mixed Operators | 4\*(3+2)%7-1 | 5 | 5.000000 | Pass |
| Case-6 | Complex Addition with Extraneous Parentheses | (((2+3)))+(((1+2))) | 8 | 8.000000 | Pass |
| Case-7 | Mixed Operators with Extraneous Parentheses | ((5\*2)-((3/1)+((4%3)))) | 6 | 6.000000 | Pass |
| Case-8 | Nested Parentheses with Exponents | (((2^(1+1))+((3-1)^2))/((4/2)%3)) | 4 | 4.000000 | Pass |
| Case-9 | Combination of Extraneous and Necessary Parentheses | (((((5-3)))\*(((2+1)))+((2\*3)))) | 12 | 12.000000 | Pass |
| Case-10 | Extraneous Parentheses with Division | ((9+6))/((3\*1)/(((2+2)))-1) | -60 | -60.000000 | Pass |
| Case-11 | Combining Unary Operator with Arithmetic Operations | +(-2)\*(-3)-((-4)/(+5)) | 6.8 | 6.800000 | Pass |
| Case-12 | Unary Negation and Addition in Parentheses | -(+1)+(+2) | 1 | 1.000000 | Pass |
| Case-13 | Negation and Addition with Negated Parentheses | -(-(-3))+(-4)+(+5) | -2 | -2.000000 | Pass |
| Case-14 | Unary Negation and Exponentiation | +2^(-3) | .125 | 0.125000 | Pass |
| Case-15 | Combining Unary Operators with Parentheses | -(+2)\*(+3)-(-4)/(-5) | -6.8 | -6.800000 | Pass |

# Environmental needs

### Hardware

Not applicable

### Software

Not applicable

### Other

Not applicable

# Special procedural requirements

Not applicable

# Intercase dependencies

Not applicable