
Group 22

Arithmetic Parser

Test Case

Version 1.2

Arithmetic Parser	Version: 1.2
Test Case	Date: 12/11/2024
05-Test-Cases	

Revision History

Date	Version	Description	Author
11/18/2024	1.0	Rough draft of the document.	Janna
12/09/2024	1.1	Continued draft of whole document.	Janna
12/11/2024	1.2	Added a test case for the keyboard input	Yoseph Ephrem

Arithmetic Parser	Version: 1.2
Test Case	Date: 12/11/2024
05-Test-Cases	

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	5
6.1.1 Hardware	5
6.1.2 Software	5
6.1.3 Other	5
7. Special procedural requirements	5
8. Intercase dependencies	5

Arithmetic Parser	Version: 1.2
Test Case	Date: 12/11/2024
05-Test-Cases	

Test Case

1. Purpose

This Test Case Specification document for the *Arithmetic Parser* defines a test case for an item that should be tested. The sections of this document are ordered in a specific sequence: Test Case Identifier, Test Item, Input specifications, Output Specifications, Environmental Needs, Special Procedural Requirements, and Intercase Dependencies.

NOTE: for sections 2, 3, 4, and 5: It is OK to use a table like the one proposed in class, also suggested on the project part 5 description.

2. Test case identifier

The Test Case ID for the UI test case is: TC-05.

The Test Case ID for the key-input test case is: TC-06

3. Test item

TC-05: $4 * (3 + 2) \% 7 - 1$

This test case includes the use of numbers on the calculator as well as the following features/operators/symbols: *, (,), +, %, and – (multiplication, parenthesis, modulo, and subtraction). These features work as specified in the Requirements Specification document and their usages are outlined in the user manual document. Both of these documents can be found in our [GitHub Repository](#).

TC-06: $(64/4)**3$

This test utilizes parenthesis to carry out PEMDAS order in the operations as well as division and powers. This case is to be tested with direct input from the keyboard but can be evaluated using the user interface buttons (" $(64 \div 4) \wedge 2$ ").

4. Input specifications

TC-05: First, open the executable file provided. To perform this test case input the following in the presented order by pressing the corresponding button on the GUI: "4", "x", "(", "3", "+", "2", ")", "%", "7", "-", "1", "=".

TC-06: to begin open the calculator by running the executable file then type the expression " $(64/4)**3$ " into the calculator and press "="

5. Output specifications

TC-05: The output specification for this test case is 5, and when executing this on the arithmetic parser, the correct output is given.

TC-06: The output for this test case is 256 and the correct value is found when using the parser.

Arithmetic Parser	Version: 1.2
Test Case	Date: 12/11/2024
05-Test-Cases	

6. Environmental needs

6.1.1 Hardware *(nothing particular for the arithmetic expression project)*

6.1.2 Software *(nothing particular for the arithmetic expression project)*

6.1.3 Other

There are no other requirements necessary for our project.

7. Special procedural requirements

8. Intercase dependencies