Boolean Algebra Calculator

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 02/05/2024 | 1.0 | Document created | Del Endecott |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 4

6.1.1 Hardware 4

6.1.2 Software 4

6.1.3 Other 4

7. Special procedural requirements 5

8. Intercase dependencies 5

# Purpose

*This Test Case Specification document for the <Project Name> defines a test case for an item that should be tested.*

*[The sections of the test case specification shall be ordered in the specified sequence. Additional sections may be included at the end. If some or all of the content of a section is in another document, then a reference to that material may be listed in place of the corresponding content. The referenced material must be attached to the test case specification or available to users of the case specification.].*

# Standard Test Cases

|  |  |
| --- | --- |
| Test Case ID | **STC01** |
| Test Case Description | Verify operators are parsed in the correct order |
| Test Data | (T | F) $ F |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC02** |
| Test Case Description | Ensure NOT operator functions correctly on parenthesized expression |
| Test Data | !(T & T) |
| Expected Results | False |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC03** |
| Test Case Description | Ensure sets of multiple parenthesized expressions are read left to right |
| Test Data | (F @ T) | (T @ F) |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC04** |
| Test Case Description | Verify AND operator only returns True when both operands are True |
| Test Data | (T $ T) & F |
| Expected Results | False |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC05** |
| Test Case Description | Ensure NOT operator functions correctly within expression |
| Test Data | !F | !T |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC06** |
| Test Case Description |  |
| Test Data | (((((T | F) & F) | (T & (T | F))) @ (T @ T)) $ (! (T | F))) |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC07** |
| Test Case Description |  |
| Test Data | ((F $ ((T | F) & (F @ (T | F)))) | (T $ (T & F))) |
| Expected Results | False |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC08** |
| Test Case Description |  |
| Test Data | (((! (T $ F)) & (T @ T)) | ((F | T) & (T $ T))) |
| Expected Results | False |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC09** |
| Test Case Description |  |
| Test Data | (((T @ T) $ (F @ T)) | ((!T) & (T | (!T)))) |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC10** |
| Test Case Description |  |
| Test Data | ((F @ T) $ (T | (F & F))) & (T & (T @ (!T))) |
| Expected Results | False |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC11** |
| Test Case Description | Ensure missing operand is handled correctly |
| Test Data | ! & T |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC12** |
| Test Case Description | Ensure unknown operator is handled correctly |
| Test Data | T ? T |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC13** |
| Test Case Description | Ensure mismatched parentheses are handled correctly |
| Test Data | ( T | ) F |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC14** |
| Test Case Description | Ensure circular logic is handled correctly |
| Test Data | T = !(T & T) |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC15** |
| Test Case Description | Ensure an empty input is handled correctly |
| Test Data | <empty input> |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC16** |
| Test Case Description | Ensure extra operators are handled correctly |
| Test Data | T &&& F |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

|  |  |
| --- | --- |
| Test Case ID | **STC17** |
| Test Case Description | Ensure unassigned variables are handled correctly |
| Test Data | X | Y |
| Expected Results | Error |
| Actual Results |  |
| Pass/Fail Status |  |

# Additional Test Cases

|  |  |
| --- | --- |
| Test Case ID | **ATC01** |
| Test Case Description | Ensure extra parentheses are ignored |
| Test Data | ((T | F)) |
| Expected Results | True |
| Actual Results |  |
| Pass/Fail Status |  |