Integration Test

# Version history

|  |  |  |  |
| --- | --- | --- | --- |
| Editor | Date | Version | Description |
| Patrick Cheung | 11 Mar 2021 | 1.0.0 | Initial release |

Table of Contents

[Version history 2](#_Toc66350830)

[Welcome page 6](#_Toc66350831)

[Case: Should be able to show welcome page when the application is launched 6](#_Toc66350832)

[User input 7](#_Toc66350833)

[Case: Should be able to throw error if user input is empty 7](#_Toc66350834)

[Case: Should be able to throw error if command is not "direct-PUSH" (i.e. not specifying the "PUSH" word) nor one of the allowed commands 8](#_Toc66350835)

[Case: Should be able to throw error if the "direct-PUSH" (i.e. not specifying the "PUSH" word) has invalid format (More than 1 part) 9](#_Toc66350836)

[Case: Should be able to throw error if the " PUSH" command has invalid format (No parameters) 10](#_Toc66350837)

[Case: Should be able to throw error if the " PUSH" command has invalid format (More than 2 parts) 11](#_Toc66350838)

[Case: Should be able to throw error if the " PUSH" command parameter is not decimal number 12](#_Toc66350839)

[Case: Should be able to throw error if the POP command has more than 1 part (it should contain only the command) 13](#_Toc66350840)

[Case: Should be able to throw error if the CLEAR command has more than 1 part (it should contain only the command) 14](#_Toc66350841)

[Case: Should be able to throw error if the ADD command has more than 1 part (it should contain only the command) 15](#_Toc66350842)

[Case: Should be able to throw error if the MUL command has more than 1 part (it should contain only the command) 16](#_Toc66350843)

[Case: Should be able to throw error if the NEG command has more than 1 part (it should contain only the command) 17](#_Toc66350844)

[Case: Should be able to throw error if the INV command has more than 1 part (it should contain only the command) 18](#_Toc66350845)

[Case: Should be able to throw error if the UNDO command has more than 1 part (it should contain only the command) 19](#_Toc66350846)

[Case: Should be able to throw error if the PRINT command has more than 1 part (it should contain only the command) 20](#_Toc66350847)

[Case: Should be able to throw error if the QUIT command has more than 1 part (it should contain only the command) 21](#_Toc66350848)

[PUSH command 22](#_Toc66350849)

[Case: Should be able to push parameter to the stack 22](#_Toc66350850)

[Case: Should be able to "direct-PUSH" (i.e. not specifying the "PUSH" word) parameter to the stack 23](#_Toc66350851)

[POP command 24](#_Toc66350852)

[Case: Should be able to pop element from the stack, where the resultant stack is not empty 24](#_Toc66350853)

[Case: Should be able to pop element from the stack, where the resultant stack is empty 25](#_Toc66350854)

[Case: Should be able to throw error if attempt to pop an empty stack 26](#_Toc66350855)

[CLEAR command 27](#_Toc66350856)

[Case: Should be able to clear all elements in stack 27](#_Toc66350857)

[Case: Should be able clear all elements in empty stack 28](#_Toc66350858)

[ADD command 29](#_Toc66350859)

[Case: Should be able to add elements 29](#_Toc66350860)

[Case: Should be able to throw error if there are not enough elements to perform addition 30](#_Toc66350861)

[MUL command 31](#_Toc66350862)

[Case: Should be able to multiply elements 31](#_Toc66350863)

[Case: Should be able to throw error if there are not enough elements to perform multiplication 32](#_Toc66350864)

[NEG command 33](#_Toc66350865)

[Case: Should be able to negate element 33](#_Toc66350866)

[Case: Should be able to throw error if attempt to perform negate on an empty stack 34](#_Toc66350867)

[INV command 35](#_Toc66350868)

[Case: Should be able to invert element 35](#_Toc66350869)

[Case: Should be able to throw error if attempt to perform inversion on an empty stack 36](#_Toc66350870)

[UNDO command 37](#_Toc66350871)

[Case: Should be able to undo "PUSH" operation, where the resultant stack is not empty 37](#_Toc66350872)

[Case: Should be able to undo "PUSH" operation, where the resultant stack is empty 38](#_Toc66350873)

[Case: Should be able to undo "POP" operation 39](#_Toc66350874)

[Case: Should be able to undo "CLEAR" operation, where the resultant stack is not empty 40](#_Toc66350875)

[Case: Should be able to undo "CLEAR" operation, where the resultant stack is empty 41](#_Toc66350876)

[Case: Should be able to undo "ADD" operation 42](#_Toc66350877)

[Case: Should be able to undo "MUL" operation 43](#_Toc66350878)

[Case: Should be able to undo "NEG" operation 44](#_Toc66350879)

[Case: Should be able to undo "INV" operation 45](#_Toc66350880)

[Case: Should be able to perform "UNDO" at the beginning, where no other operations have been executed yet 46](#_Toc66350881)

[PRINT command 47](#_Toc66350882)

[Case: Should be able to all elements 47](#_Toc66350883)

[Case: Should be able to show message if the stack is empty 48](#_Toc66350884)

[QUIT command 49](#_Toc66350885)

[Case: Should be able to quit application 49](#_Toc66350886)

[Logging 50](#_Toc66350887)

[Case: Should be able to log error logs related to program run 50](#_Toc66350888)

[Case: Should not show the logs in the console 51](#_Toc66350889)

# Welcome page

## Case: Should be able to show welcome page when the application is launched

Text

Description automatically generated

# User input

## Case: Should be able to throw error if user input is empty

Graphical user interface, text, application

Description automatically generated

## Case: Should be able to throw error if command is not "direct-PUSH" (i.e. not specifying the "PUSH" word) nor one of the allowed commands

Graphical user interface, text

Description automatically generated

## Case: Should be able to throw error if the "direct-PUSH" (i.e. not specifying the "PUSH" word) has invalid format (More than 1 part)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the " PUSH" command has invalid format (No parameters)

Graphical user interface, application

Description automatically generated

## Case: Should be able to throw error if the " PUSH" command has invalid format (More than 2 parts)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the " PUSH" command parameter is not decimal number

Graphical user interface, application

Description automatically generated

## Case: Should be able to throw error if the POP command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the CLEAR command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the ADD command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the MUL command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the NEG command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the INV command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the UNDO command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the PRINT command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

## Case: Should be able to throw error if the QUIT command has more than 1 part (it should contain only the command)

Graphical user interface, text

Description automatically generated with medium confidence

# PUSH command

## Case: Should be able to push parameter to the stack

Text

Description automatically generated

## Case: Should be able to "direct-PUSH" (i.e. not specifying the "PUSH" word) parameter to the stack

Text

Description automatically generated

# POP command

## Case: Should be able to pop element from the stack, where the resultant stack is not empty

Text

Description automatically generated

## Case: Should be able to pop element from the stack, where the resultant stack is empty

Text

Description automatically generated

## Case: Should be able to throw error if attempt to pop an empty stack

Graphical user interface, text

Description automatically generated with medium confidence

# CLEAR command

## Case: Should be able to clear all elements in stack

Text

Description automatically generated

## Case: Should be able clear all elements in empty stack

Text

Description automatically generated

# ADD command

## Case: Should be able to add elements

Text

Description automatically generated

## Case: Should be able to throw error if there are not enough elements to perform addition

Graphical user interface, application

Description automatically generated

# MUL command

## Case: Should be able to multiply elements

Text

Description automatically generated

## Case: Should be able to throw error if there are not enough elements to perform multiplication

Text

Description automatically generated

# NEG command

## Case: Should be able to negate element

Graphical user interface, application

Description automatically generated

## Case: Should be able to throw error if attempt to perform negate on an empty stack

Graphical user interface, application

Description automatically generated

# INV command

## Case: Should be able to invert element

Graphical user interface, application

Description automatically generated

## Case: Should be able to throw error if attempt to perform inversion on an empty stack

Graphical user interface, application

Description automatically generated

# UNDO command

## Case: Should be able to undo "PUSH" operation, where the resultant stack is not empty

Text

Description automatically generated

## Case: Should be able to undo "PUSH" operation, where the resultant stack is empty

Text

Description automatically generated

## Case: Should be able to undo "POP" operation

Text

Description automatically generated

## Case: Should be able to undo "CLEAR" operation, where the resultant stack is not empty

Text

Description automatically generated

## Case: Should be able to undo "CLEAR" operation, where the resultant stack is empty

Text

Description automatically generated

## Case: Should be able to undo "ADD" operation

Text

Description automatically generated

## Case: Should be able to undo "MUL" operation

Text

Description automatically generated

## Case: Should be able to undo "NEG" operation

Text

Description automatically generated

## Case: Should be able to undo "INV" operation

Text

Description automatically generated

## Case: Should be able to perform "UNDO" at the beginning, where no other operations have been executed yet

Text

Description automatically generated

# PRINT command

## Case: Should be able to all elements

Text

Description automatically generated

## Case: Should be able to show message if the stack is empty

Graphical user interface, text, application

Description automatically generated

# QUIT command

## Case: Should be able to quit application

Graphical user interface, text, application

Description automatically generated

# Logging

## Case: Should be able to log error logs related to program run

Graphical user interface, text, application, website

Description automatically generated

## Case: Should not show the logs in the console

Graphical user interface, application

Description automatically generated