Portfolio Question 1.7

Calculator Documentation

GitHub Link: https://github.com/DarylP45/Calculator

Contents

[Coding Convention and Comments 3](#_Toc514419003)

[Arithmetic 3](#_Toc514419004)

[Algebra 4](#_Toc514419005)

[Trigonometry 4](#_Toc514419006)

[Calculator 5](#_Toc514419007)

[Data Structures 9](#_Toc514419008)

[Algorithms ( Pseudo Code ) 10](#_Toc514419009)

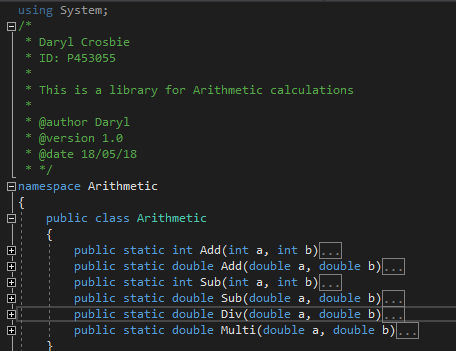
[Error Handling 12](#_Toc514419010)

[Testing Procedure 12](#_Toc514419011)

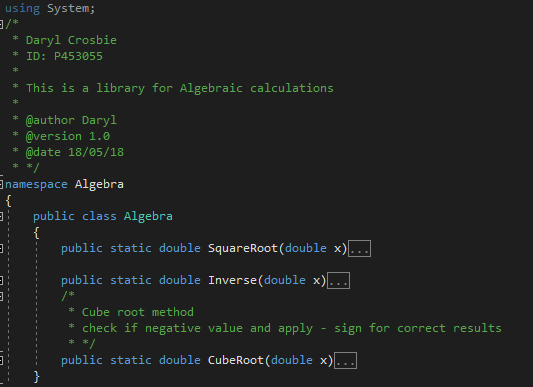
[Recommendations 13](#_Toc514419012)

# Coding Convention and Comments

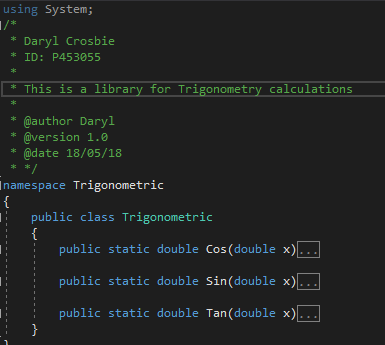
## Arithmetic



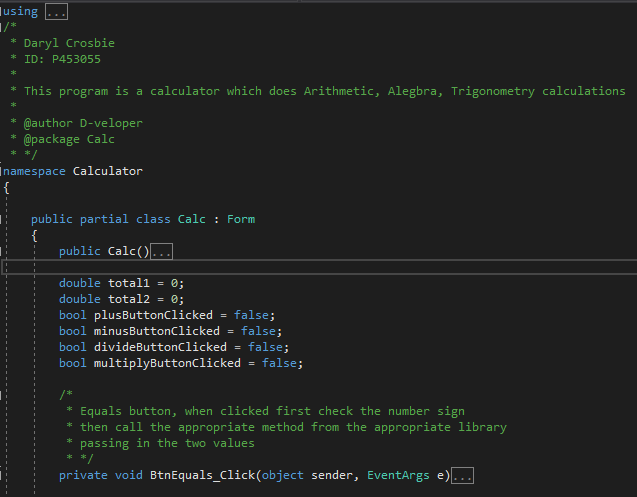
## Algebra

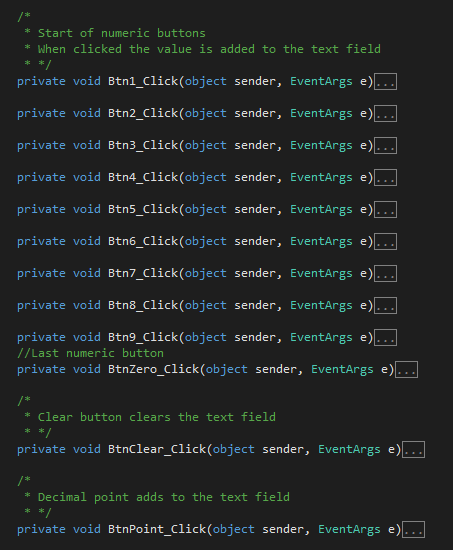


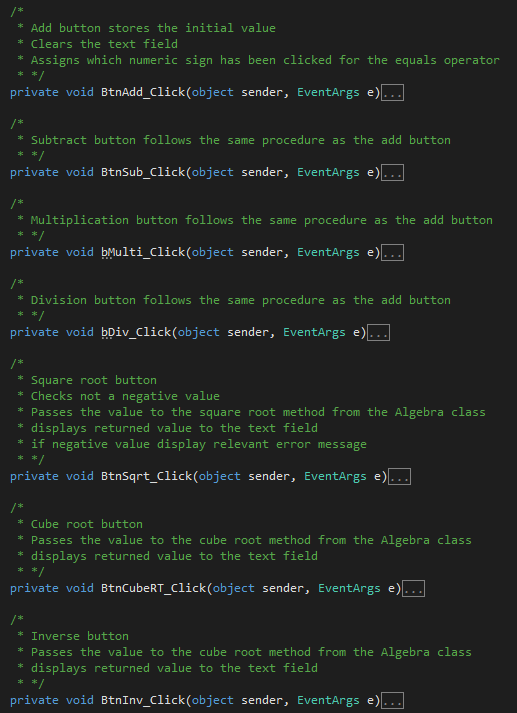
## Trigonometry

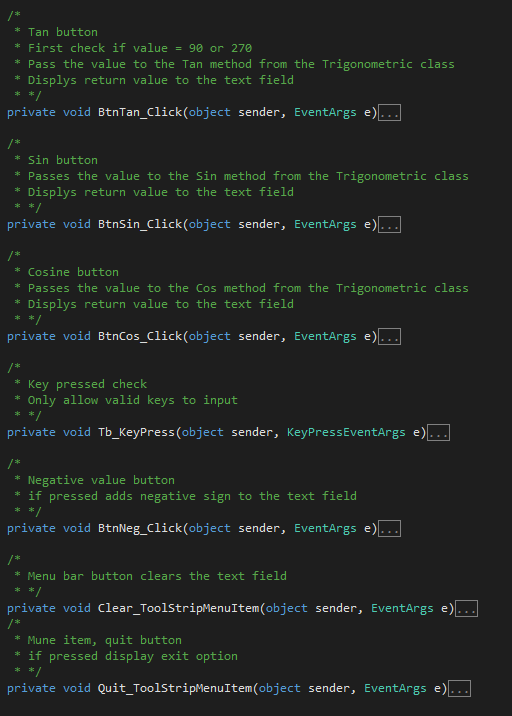


## Calculator









# Data Structures

|  |  |  |
| --- | --- | --- |
| Name | Type | Purpose |
| Total1 | Double | Stores the first input value |
| Total2 | Double | Stores the value from the arithmetic calculation |
| plusButtonClicked | Bool | Check for arith op select |
| minusButtonClicked | Bool | Check for arith op select |
| divideButtonClicked | Bool | Check for arith op select |
| multiplyButtonClicked | Bool | Check for arith op select |
| Num | Double | Stores a parsed string |
| Root | Double | Store cube root result |
| Roundedroot | Double | Stores the rounded root value |

# Algorithms ( Pseudo Code )

Equals button

Check ( operator )

Values -> appropriate method

Display result

Reset total1

Arithmetic buttons pressed

Store value

Set boolean value

Clear the text field

Square Root button

Parse and store the input

if (value not negative)

value -> Square root method

Display returned value

else

Display error message

Cube Root / Inverse button

Parse and store the value

value -> algebra method

Display return value

Tan button

Parse and store the input value

if (value = 90 or 270)

Display error message

Else pass value to Tan method

Display return value

Cos / Sin button

Parse and store input value

value -> trigonometry method

Display return value

Square Root Method ( double x )

Return Sqrt ( x )

Inverse Method ( double x )

Return 1 / x

Cube Root Method ( double x)

If ( value contains “-“ )

Root = Pow ( -x, (1.0 / 3.0))

roundedRoot = Round ( Root )

if ( Abs (roundedRoot – root ) < 1e-10 ) )

return –roundedRoot

else

return -root

else

Root = Pow ( x, ( 1.0 / 3.0 ) )

roundedRoot = Round ( Root )

if ( Abs (roundedRoot – root ) < 1e-10 ) )

return roundedRoot

else

return root

Cos / Sin Method (double x)

Return Round ( Cos/Sin ( PI \* ( x / 180.0 ) ) ) , 3 )

Tan Method (double x)

Return Round ( Tan ( x \* ( PI /180.0 ) ) ) , 3 )

# Error Handling

Input text field key character check applied to only allow valid input.

Check arithmetic booleans for all false first when equals button pressed.

Tan button check for 90 or 270.

Cube root method checks for negative value.

# Testing Procedure

Description: The calculator should be able to do simple arithmetic, algebraic, and trigonometric calculation. The user can enter values and select an operator.

Testing steps:

* Enter values to the text field ( a – z ) ( 0 – 9 ) ( 0.01 ).
* Use each calculation button on a range of numeric values and including negative values and decimal values.
* Use menu option to clear and quit application

Expected Result: Calculations will return with correct value, and if invalid will display an appropriate error message.

**Actual result: Returned correct values, and displayed error messages when necessary.  
Pass/Fail: PASS**

# Recommendations

An added display field should be added to show all values and operators being pressed.

A history menu should be added containing the last 10 calculations.

As this calculator only handles Arithmetic, Algebraic, and Trigonometric calculations a wide range of other function should be add to boost the power of this calculator. More menus can be added for different fields of functions, in this way we can keep the main display simple and easy to read.