./

GENESIS - Learning Outcome & Mini-project Summary Report



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be Approved** | **Remarks/Revision Details** |
| 1. | 07-12-20 | Bandi Ramesh |  | Pagala Prithvi Sekhar |  |
| 2. | 07-12-20 | Aruljothi C |  | Pagala Prithvi Sekhar |  |
| 3. | 07-12-20 | John Wesley D |  | Pagala Prithvi Sekhar |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Details**

Contents

[Contents 3](#_Toc58233485)

[Table of Figures 4](#_Toc58233486)

[Table of Tables 4](#_Toc58233487)

[**Introduction** 5](#_Toc58233488)

[**Objectives & Requirements** 5](#_Toc58233489)

[Costing 5](#_Toc58233490)

[Ageing 6](#_Toc58233491)

[4W1H Question 6](#_Toc58233492)

[High level requirements 7](#_Toc58233493)

[Low level requirements 7](#_Toc58233494)

[SWOT Analysis 8](#_Toc58233495)

[Design 9](#_Toc58233496)

[Test plan 13](#_Toc58233497)

Implementation summary 14

[Github link 14](#_Toc58233498)

[Github dashboard 14](#_Toc58233499)

Summary 15

Reference 15

## Table of Figures

[Figure 1- Costing chart for calculator 5](#_Toc53222229)

[Figure 2- Ageing chart for calculator 6](#_Toc53222230)

[Figure 3- Component diagram for calculator 9](#_Toc53222231)

[Figure 4- Use case diagram for calculator 9](#_Toc53222232)

[Figure 5- Use case Diagram for calculator 10](#_Toc53222233)

[Figure 6- UML diagram for calculator 10](#_Toc53222234)

[Figure 7-. Sequence diagram for calculator 11](#_Toc53222235)

[Figure 8- State diagram for calculator 12](#_Toc53222236)

## Table of Tables

[Table 1- High level requirements 7](#_Toc53222229)

[Table 2- Low level requirements 7](#_Toc53222230)

[Table 3- SWOT analysis 8](#_Toc53222231)

[Table 4- Test plan 13](#_Toc53222232)

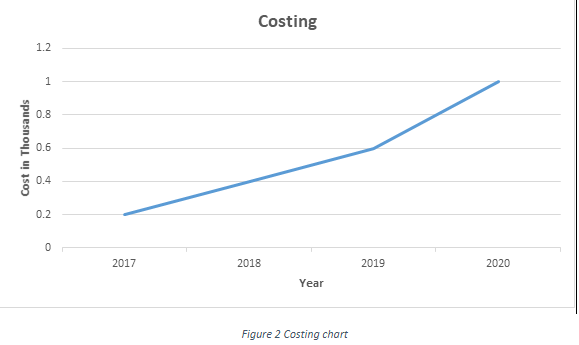
MINIPROJECT -CALCULATOR

## **Introduction**

A calculator is a device that performs arithmetic operations on numbers. The simplest calculators can do only addition, subtraction, multiplication, and division. More sophisticated calculators can handle exponential operations, roots, logarithms, trigonometric functions, and hyperbolic functions. Internally, some calculators actually perform all of these functions by repeated processes of addition. The Casio Computer Company developed the first electronic calculator in 1957. Since then, calculators have come in many sizes, and are also built into most operating systems on computers, smartphones, and tablets. A calculator is a person who performs arithmetic or other mathematical calculations.

## **Objectives & Requirements**

### Costing

****

#### Figure1. Costing chart for calculator

### Ageing

#### Figure2. Ageing chart for calculator

### 4W1H Question

**What?**

A calculator is a small [hand-held computer](https://www.webopedia.com/TERM/H/hand_held_computer.html) that performs mathematical calculations. Some calculators even permit simple [text](https://www.webopedia.com/TERM/T/text.html) editing and [programming](https://www.webopedia.com/TERM/P/program.html).

**When?**

People do calculations usually use calculators in everyday life to save their time and for accurate answers.

**Where?**

Many times, in scientific and mathematical calculations involving complex operations with complicated numbers the calculation is not feasible because it will take a lot of time and there are many chances of errors when done manually.

**Why?**

To make calculations easier.

**How?**

The memory chips inside the calculator contain thousands or millions of bytes program code that allows the calculator to do work.

### High level requirements

|  |  |
| --- | --- |
| **H1: Arithmetic function** | This is a function which related to the basic operations. |
| **H2: Temperature conversion** | This is the key component that implements the functions related to the temperature conversion in a calculator. |
| **H3: Math function** | This is the function which is responsible for the different types of operation in math function. |

Table 1:High level requirements

### Low level requirements

|  |  |  |
| --- | --- | --- |
| **H1: Arithmetic function** | **H2:Temperature conversion** | **H3: Math function** |
| **L1:Basic operation-** Basic operations like addition, subtraction, multiplication and division takes place. | **L2:Differentconversion**-All the conversions of Kelvin, Celsius and Fahrenheit. | **L3:Basic math functions-** In this basic math functions like squareroot,power,exponential etc., takes place. |

Table 2:Low level requirements

### SWOT Analysis

|  |  |  |  |
| --- | --- | --- | --- |
| **Strength** | **Weakness** | **Opportunity** | **Threats** |
| * Can solve complicated problems easily * Give more accurate results * Time saving * Avoids boredom in counting process | * People will become lazy * Calculator limits the knowledge of user * People will be unable to memorize the process of problem solving | * The future value   calculator can be  used to determine  future value,or Fv infinancing | * Cheating. The availability of graphic   calculators has made it easier for students to cheat during their tests. |

Table 3:SWOT analysis

### 

### Design

**Component diagram**

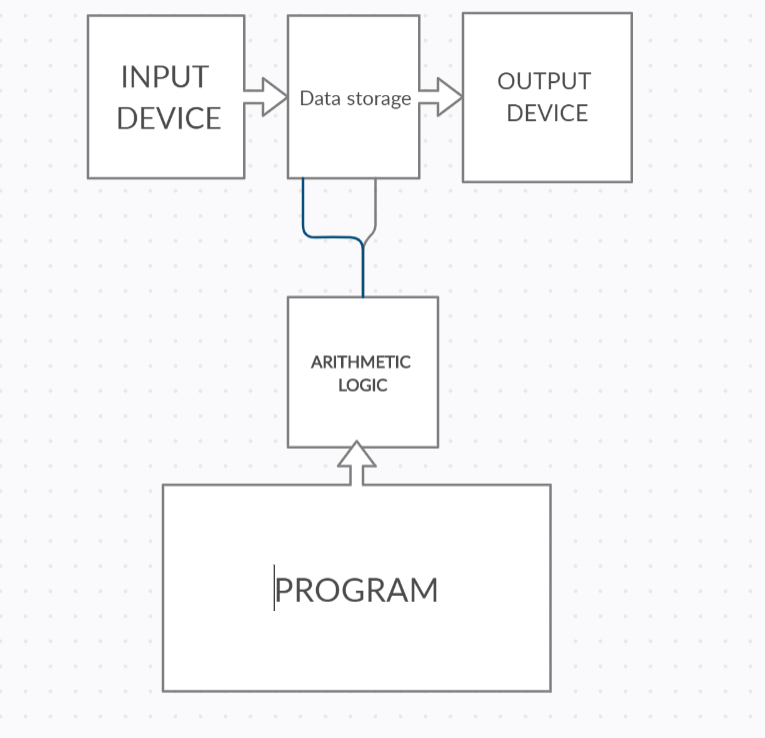


Figure 3. Component diagram for calculator

**Use case diagram**

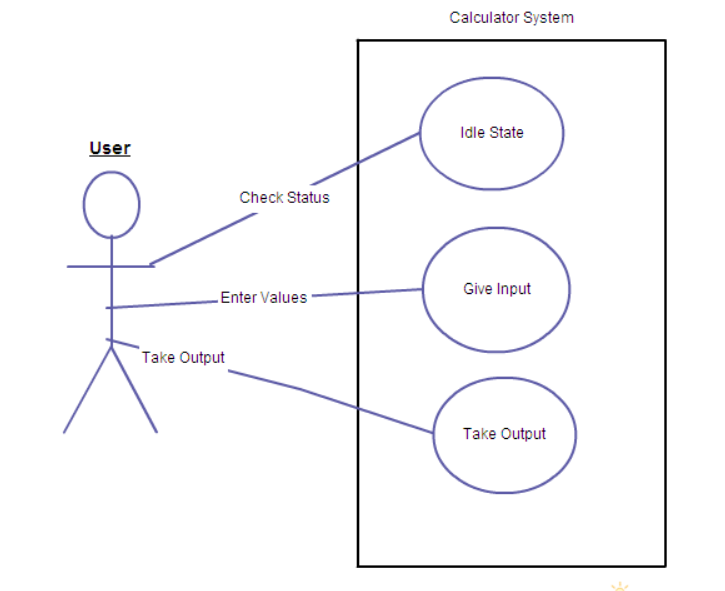


Figure 4. Use case diagram for calculator

**Use case diagram**

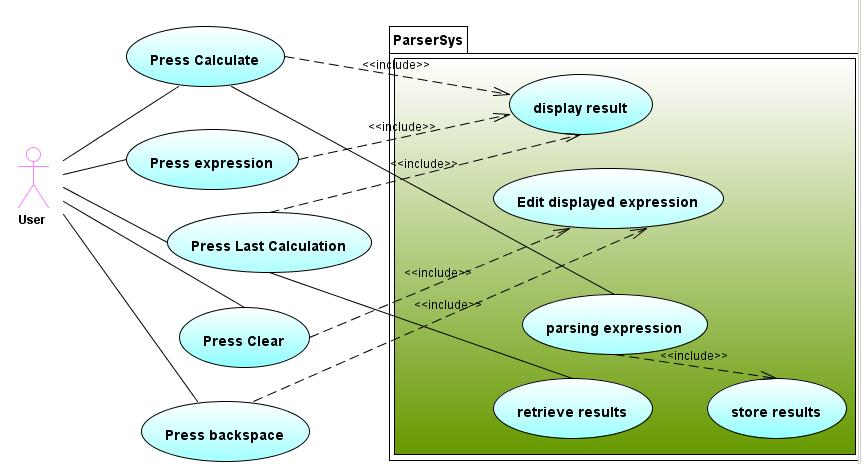


Figure 5. Use case Diagram for calculator

**UML diagram**

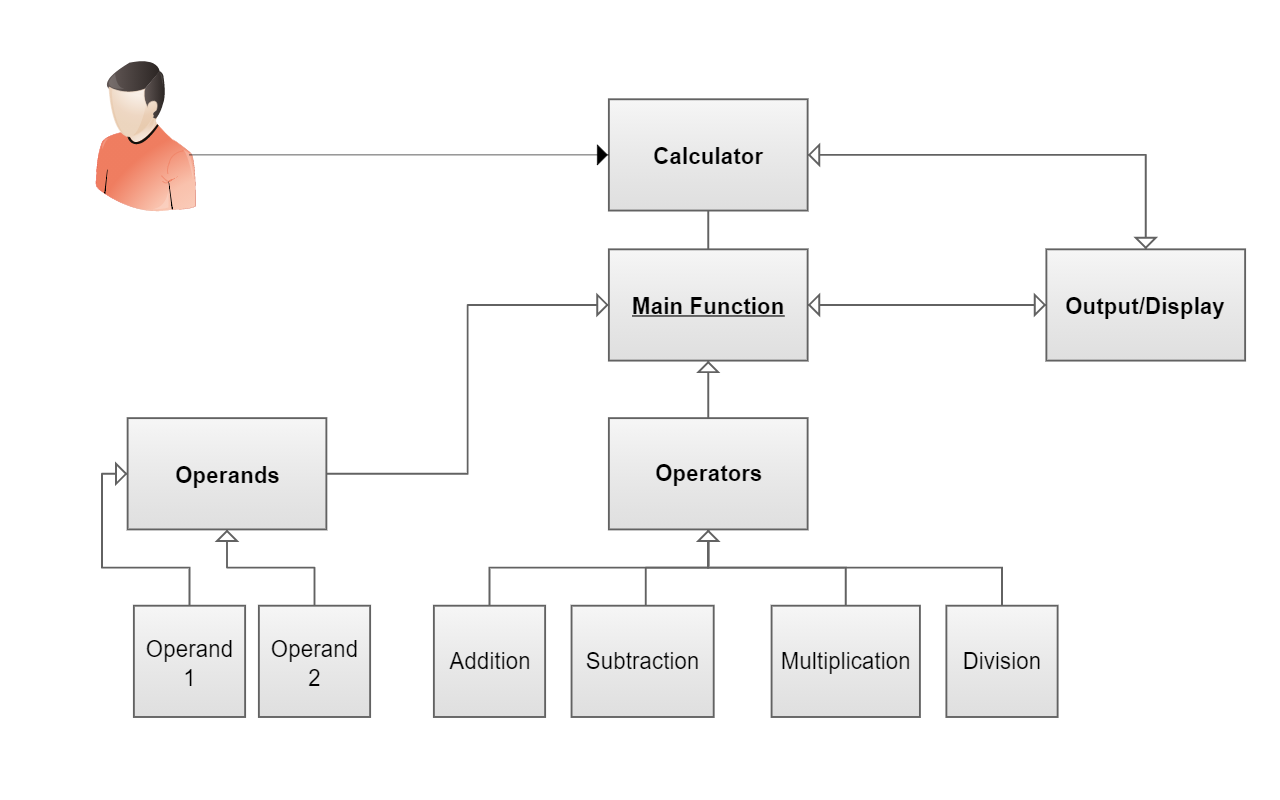


Figure 6. UML diagram for calculator

**Sequence diagram**

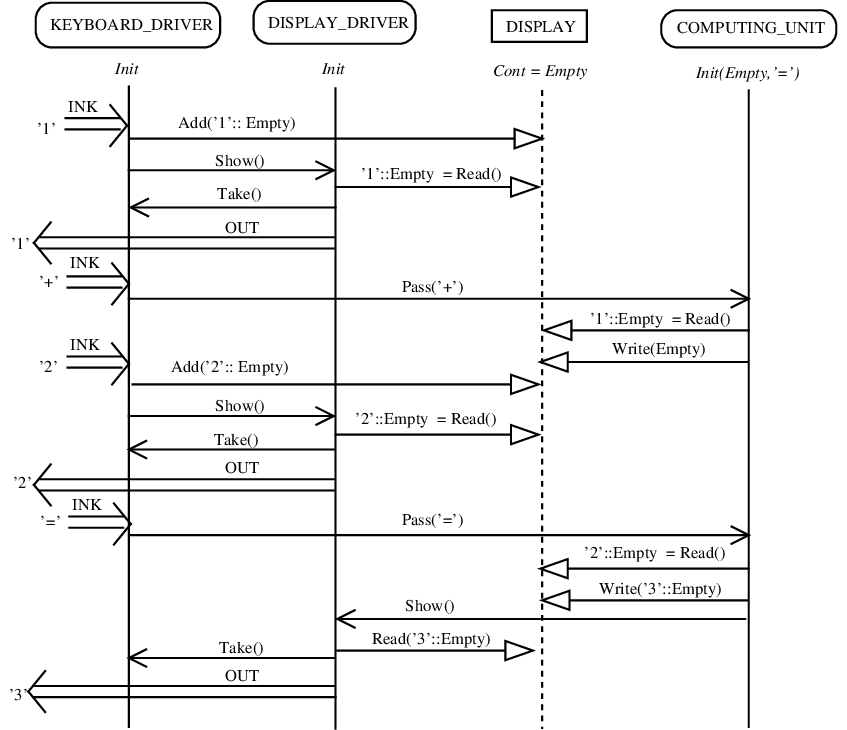


Figure 7. Sequence diagram for calculator

**State diagram**

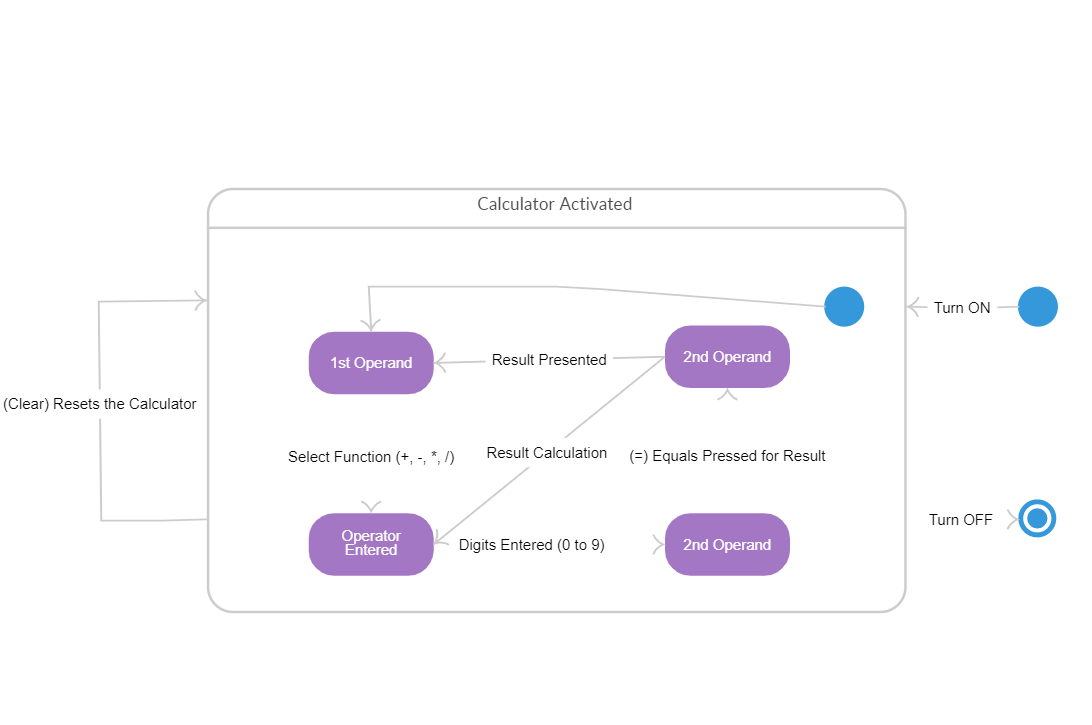


Figure 8. State diagram for calculator

### Test plan

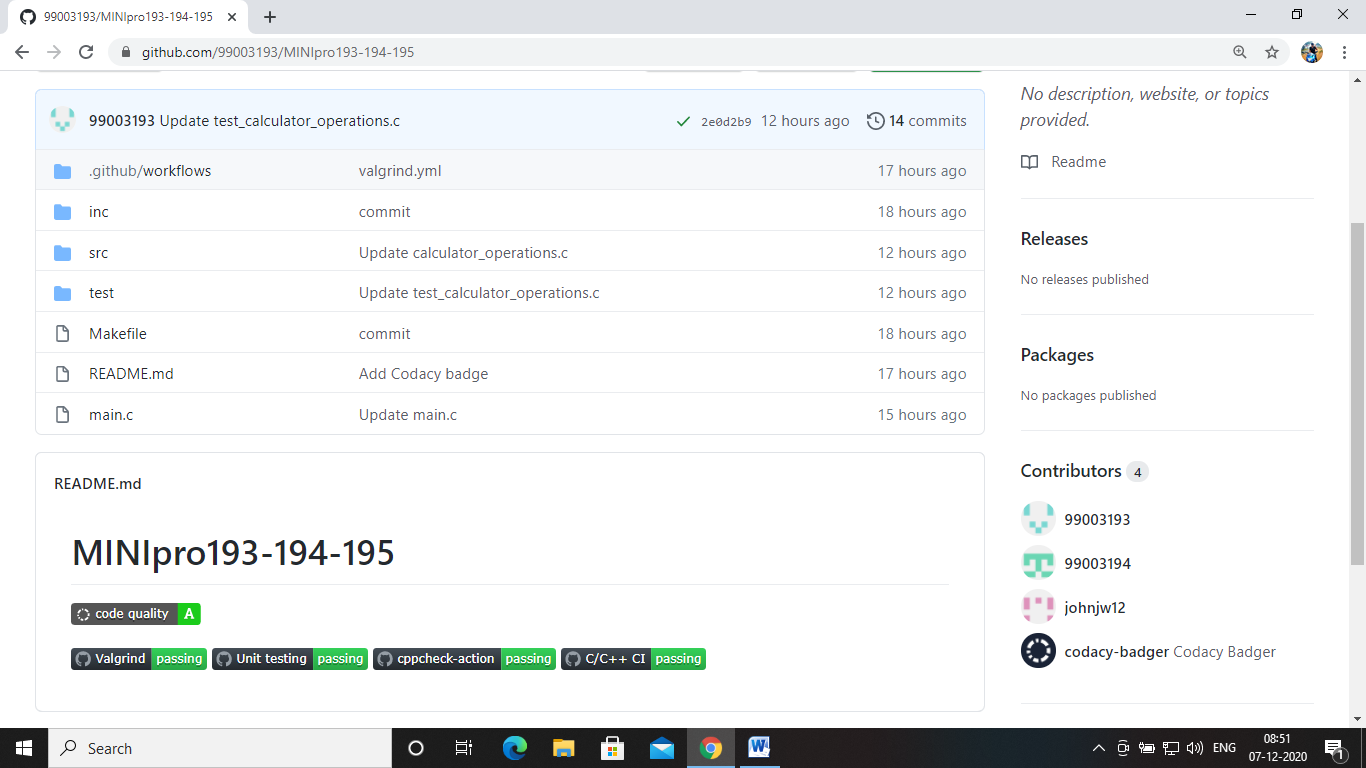
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SL.NO** | **TEST**  **ID** | **DESCRIPTION** | **Expected input** | **Expected**  **output** | **Actual output** |
| **1.** | TC1 | Check if it starts via specific means like from searching for calculator in search bar | Give the input | It should accept |  |
| **2.** | TC2 | Check the addition of numbers. | Anynumbers(positive,  Negative, any data type) | It should accept |  |
| **3.** | TC3 | Check the subtraction of numbers. | Any numbers(positive,  Negative, any data type) | It should accept |  |
| **4.** | TC4 | Check the multiplication of numbers. | Any numbers(positive,  Negative, any data type) | It should accept |  |
| **5.** | TC5 | Check the division i numbers. | Any numbers(positive,  Negative, any data type) | It should accept |  |
| **6.** | TC6 | Check the division of a number by zero. | Any number(positive ,negative) | It should accept |  |
| **7.** | TC7 | Check if the functionality using BODMAS | Enter Correct User input | It should accept |  |
| **8.** | TC8 | Check if the sin, cos, tan and cos is operational using the keys. | Give correct input keys | It should accept |  |
| **9.** | TC9 | Check if the x-1, x!,|x|,x^y and f(x) is operational | Enter Correct user input | It should accept |  |
| **10.** | TC10 | Check if the log key is operational | Give correct key | It should accept |  |
| **11.** | TC11 | Check if the factorial key is working | Give correct key | It should accept |  |
| **12.** | TC12 | Check if the Square root is working | Give correct key | It should accept |  |
| **13.** | TC13 | Check if the number is ODD or EVEN is working | Give any number(positive,negative) | It should accept |  |

Table 4: Test plan

## **Implementation Summary**

Github link: https://github.com/99003193/MINIpro193-194-195

Github dashboard:



### Summary

**My individual contribution**

* + - * 1.I done arithmetic function part code
      * 2.cpp check
      * 3.unit test
      * 4.In design part I done 2 design diagram for arithmetic functions
      * 5.For document preparation I done costing and ageing and high level and low level requirements.

**Challenges faced and how I overcome**

### This was the first time I did so many documents and I faced some challenges while preparing diagrams for design part.

* I faced some challenges for preparing test plan and also I faced some challenges while uploading code and folders directly push it to the github and in unit test also initially I faced little bit difficult.
* I refer some websites and I asked my friends to clarify my doubts.

**Appreciation**

I would like to appreciate my teammateBandi Ramesh(99003193) who helped me a lot for how to directly push the code to github and he helped me to clear errors.

**Reference**

* <https://fresh2refresh.com/c-programming/c-programs/c-code-for-calculator-application/>
* <https://www.geeksforgeeks.org/working-on-git-bash/>
* <https://www.guru99.com/unit-testing-guide.html>