MINI PROJECT REPORT ON

C -PROGRAMMIMG

Module: C-Programming
Project title : Simple-Calculator

Submitted by
Sujata Naikar

https://github.com/sujatann/M1_Simple-Calculator_Utility.git

[PS:99007669]

AIM: To design a simple calculator using c programming.

OBJECTIVES

- To make a user interactive program , a simple calculator.
- Two numbers are accepted to run the operation for addition, substraction, multiplication and division, and only one number is asked as an input for the operations such as to find out squares and cubes.
- Exceptions have been specified in the code .

ABSTRACT

The project that I would be doing will mainly stress on performing simple mathematic calculations that are required while adding, substracting, dividing, multiplying any two numbers and gives squares and cubes of any particular number that you ask for. This project consists of a program which takes an arithmetic operator +, -, *, / and two operands from the user. Then, it performs the calculation on the two operands depending upon the operator entered by the user.

SOFTWARES REQUIRED

- Visual studio code, github, ubuntu-wsl.

There is a folder as milestone-1,which in short gives idea about what all is to be covered in carrying out the project.

❖ Milestone-1

- Folder structure.
- Idea
- Requirements
- Design
- Test plan

❖ ISSUE THAT WAS CREATED AT THE STARTING OF ANALYSING THE PROBLEM STATEMENT.

1.Capture a well processed idea.
2.Requirements for the project.
-low level and high level
3.Design(behavioural and structural)
4.Test plan and Validation.

1. CERTIFICATES INCLUDE:

1.1 Sololearn.

1.2 Cisco NDG Linux.

Dear Sujata Naikar,

Congratulations on completing the NDG Linux Unhatched course in the Cisco Networking Academy. This letter documents you have successfully completed the NDG Linux Unhatched course, which provides an introduction to the Linux command line. Linux is everywhere! As the reach of Linux continues to grow, knowledge of Linux is a core skill for all IT professionals. By completing this course, you have gained a better understanding of Linux.

If you decide to pursue additional knowledge of Linux consider:

NDG LINUX ESSENTIALS

This course is the perfect next step for beginners looking to expand their skills and knowledge of Linux. This full-semester course can be delivered as instructor-led training or as a self-paced learning experience. The NDG Linux Essentials course is designed to prepare you for the Linux Professional Institute Linux Essentials Professional Development Certificate.

NDG LINUX SERIES

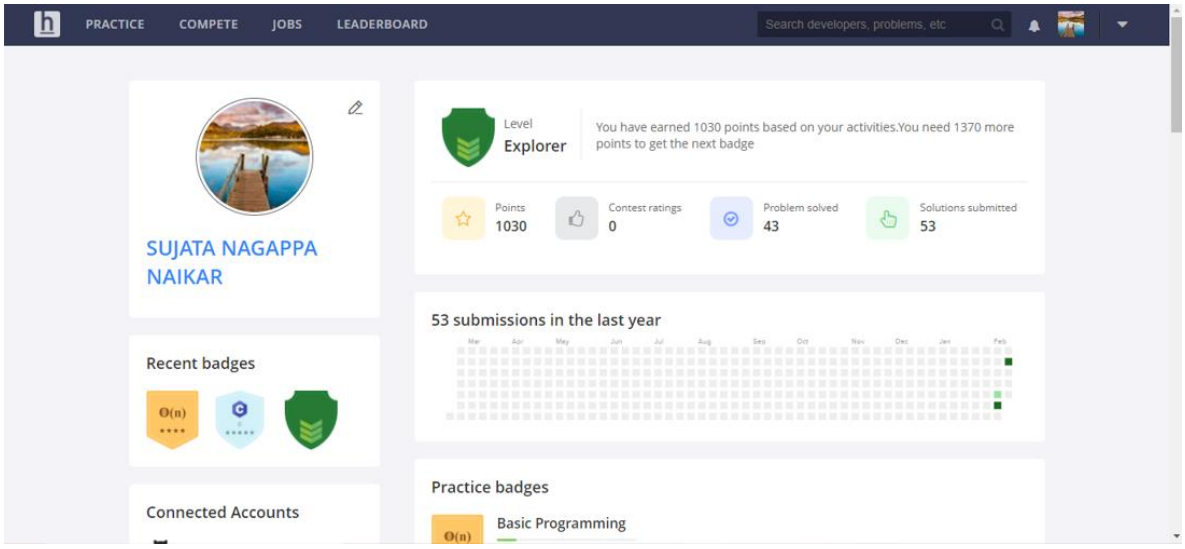
This certification level series offers beginners more rigorous in-depth coverage. The NDG Introduction to Linux I and NDG Introduction to Linux II courses focus on the basic Linux system administration skills needed in preparation for the Linux Professional Institute LPIC-1 certification.

Again, congratulations and we wish you continued success!

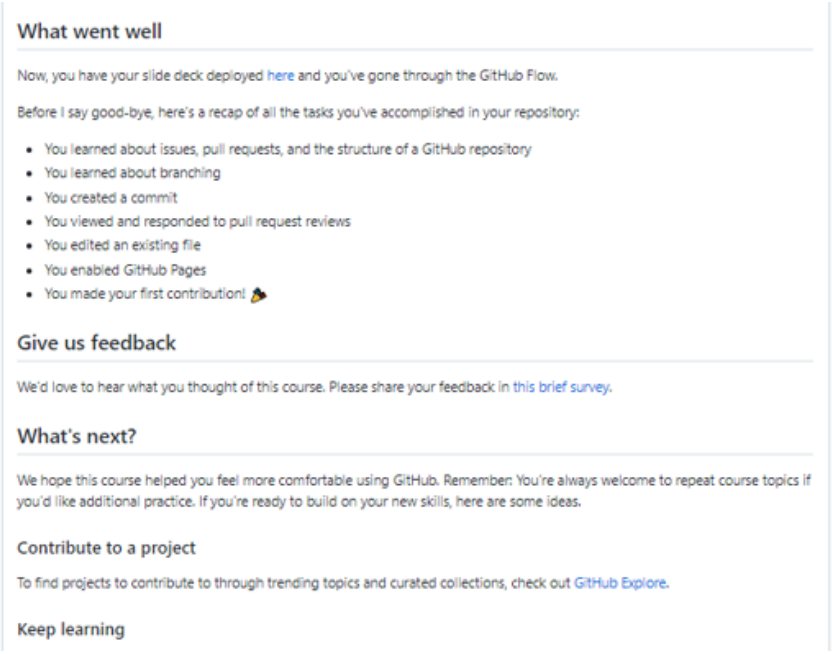
Sincerely,
The NDG Team

Date 2 Feb 2022

1.3 Screenshot of HackerEarth points.



1.4 Screenshot of Github learning.



What went well

Now, you have your slide deck deployed here and you've gone through the GitHub Flow.

Before I say good-bye, here's a recap of all the tasks you've accomplished in your repository:

- You learned about issues, pull requests, and the structure of a GitHub repository
- You learned about branching
- You created a commit
- You viewed and responded to pull request reviews
- You edited an existing file
- You enabled GitHub Pages
- You made your first contribution!

Give us feedback

We'd love to hear what you thought of this course. Please share your feedback in this brief survey.

What's next?

We hope this course helped you feel more comfortable using GitHub. Remember: You're always welcome to repeat course topics if you'd like additional practice. If you're ready to build on your new skills, here are some ideas.

Contribute to a project

To find projects to contribute to through trending topics and curated collections, check out GitHub Explore.

Keep learning

2.Requirements

High level Requirements

- Able to perform operations such as,

1. Addition.
2. Substraction.
3. Multiplication.
4. Division.
5. Square.
6. Cube.

Low level Requirements

1. Taking the two numbers to do any operation.
2. A code is written where different commands are used for different operations.
3. In case of squaring and cubing a number take only one number as an input.
4. The answer must be displayed on the screen.

SWOT Analysis

1. Strength

- Capable of adding, substracting, dividing, multiplying any two numbers and gives squares and cubes of any particular number that you ask for.

2. Weakness

- As the main intension is to work with two numbers provided more will not be supported.

3. Opportunities

- In the basic step to make someone understand the simple operations of mathematics it is most useful one.

4. Threat

- Will not ask for more than two numbers.

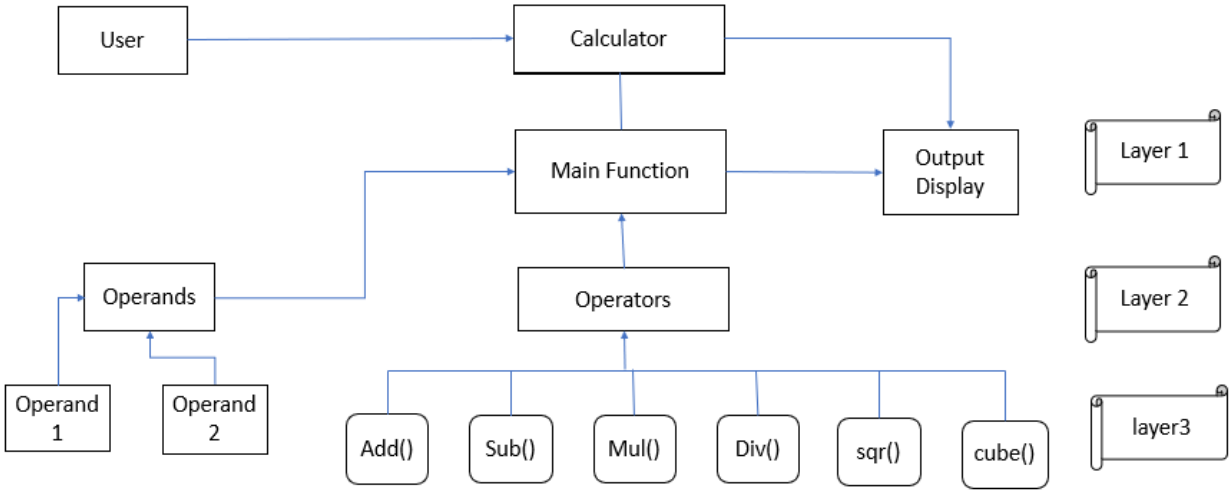
❖ 5W’s and 1 H

- When to use is in need of the requirement of basic calculations.
- Where: can be in schools, working place, counters etc.
- Why can be stated in case to know the exact calculations.
- Who is someone who wants to take use of it and find solution for their problems.
- What is by using numbers in the calculator.
- H for How, is by taking two numbers as input and carrying out any operation of the choice.

3.Architecture.

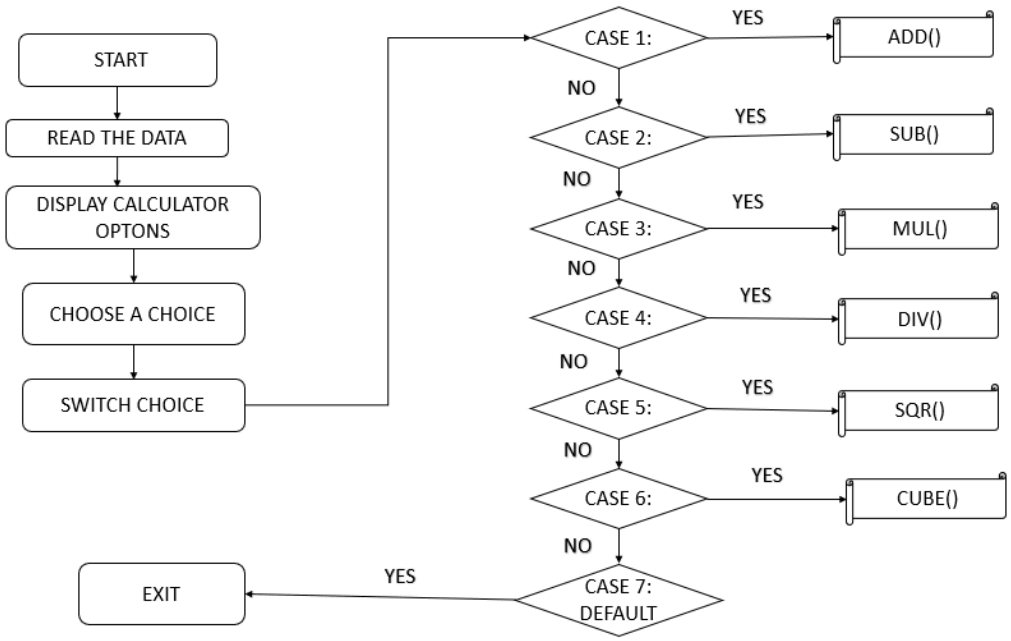
Design

3.1Structural diagrams

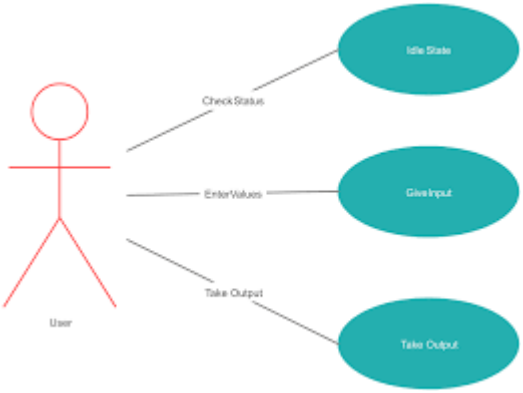


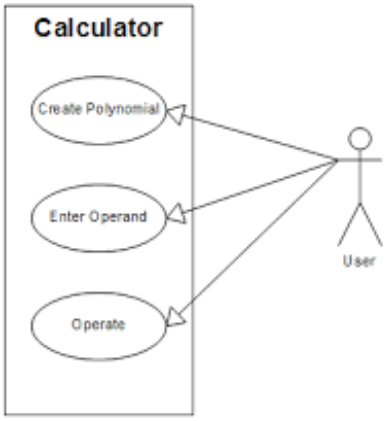
3.2 Behavioural diagrams

- Flowcharts



- Use case diagrams





MODELS:

4.IMPLEMENTATION.

CODE FOR SIMPLE CALCULATOR

```
//simple computer:add,sub,mul,div,sqr,cube
// progarm to obtain some basic operations used in simple calculators
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
int choose;
long no1, no2, m;

printf(" select any one from the  options  below:"

"\n1 = Addition\n"

"\n2 = Subtraction\n"

"\n3 = Multiplication\n"

"\n4 = Division\n"

"\n5 = Squares\n"

"\n6 = Cubes\n"
```

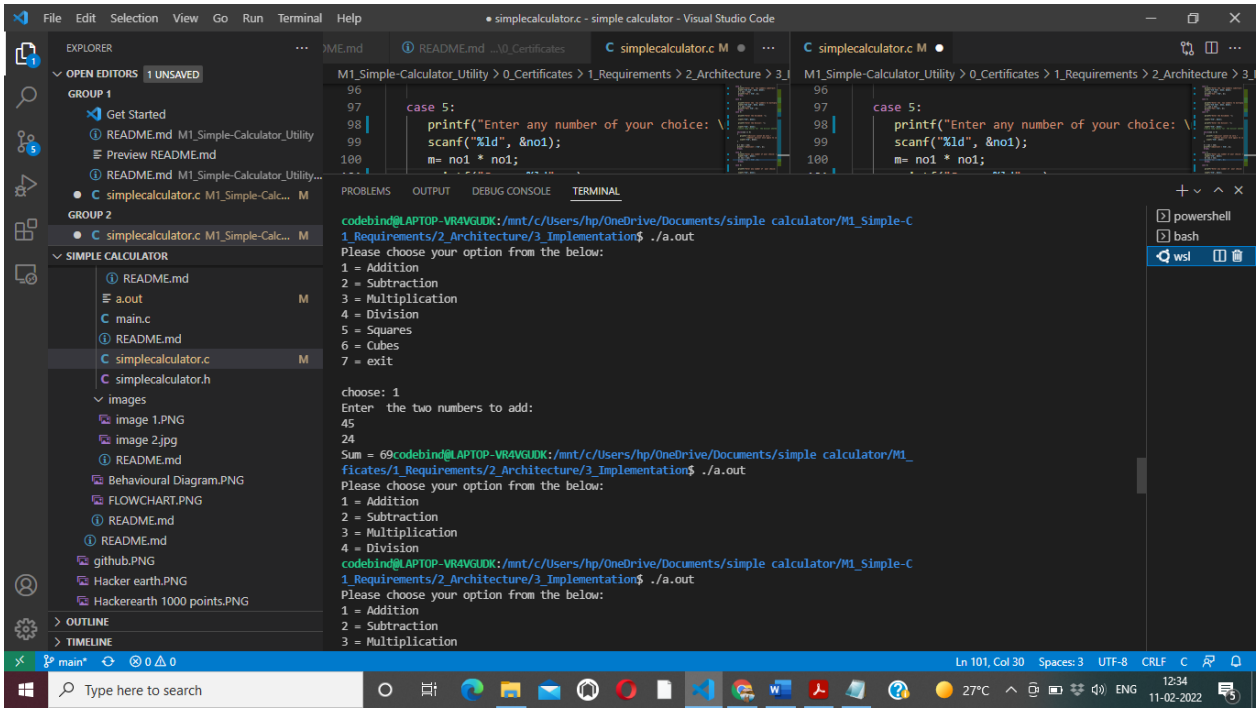
```
"\n7 = exit\n"


"\n\nchoose: ");


scanf("%d", &choose);
//while loop check whether the choose option  is in the given range
while(choose< 1 || choose > 7)
{
printf("\nchoose the above mentioned option."
"\nChoose: ");
scanf("%d", &choose);
}
switch (choose)
{


case 1:
printf(" Please enter  any  two numbers to add: \n");


scanf("%ld %ld", &no1, &no2);
m = no1 + no2;
printf("Sum = %ld", m);
break;


case 2:
```

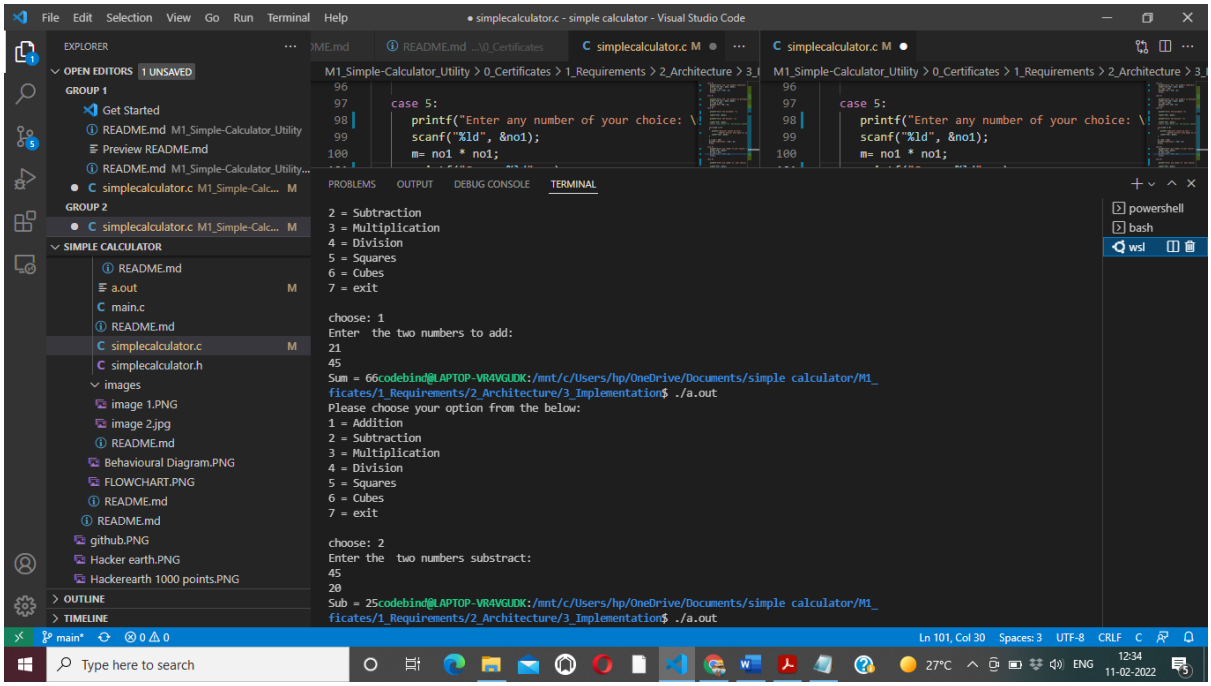
```
printf("Please enter any  two numbers  to substract: \n");
scanf("%ld %ld", &no1, &no2);
m = no1 - no2;
printf("Sub = %ld", m);
break;

case 3:

printf("Please enter any two numbers to multiply: \n");
scanf("%ld %ld", &no1, &no2);
m = no1 * no2;
printf("Pdt= %ld", m);
break;
case 4:

printf("Enter the Dividend: ");

scanf("%ld", &no1);

printf("Enter the Divisor: ");

scanf("%ld", &no2);
//while loop checks for  the divisor whether it is zero or not

while(no2 == 0)
```

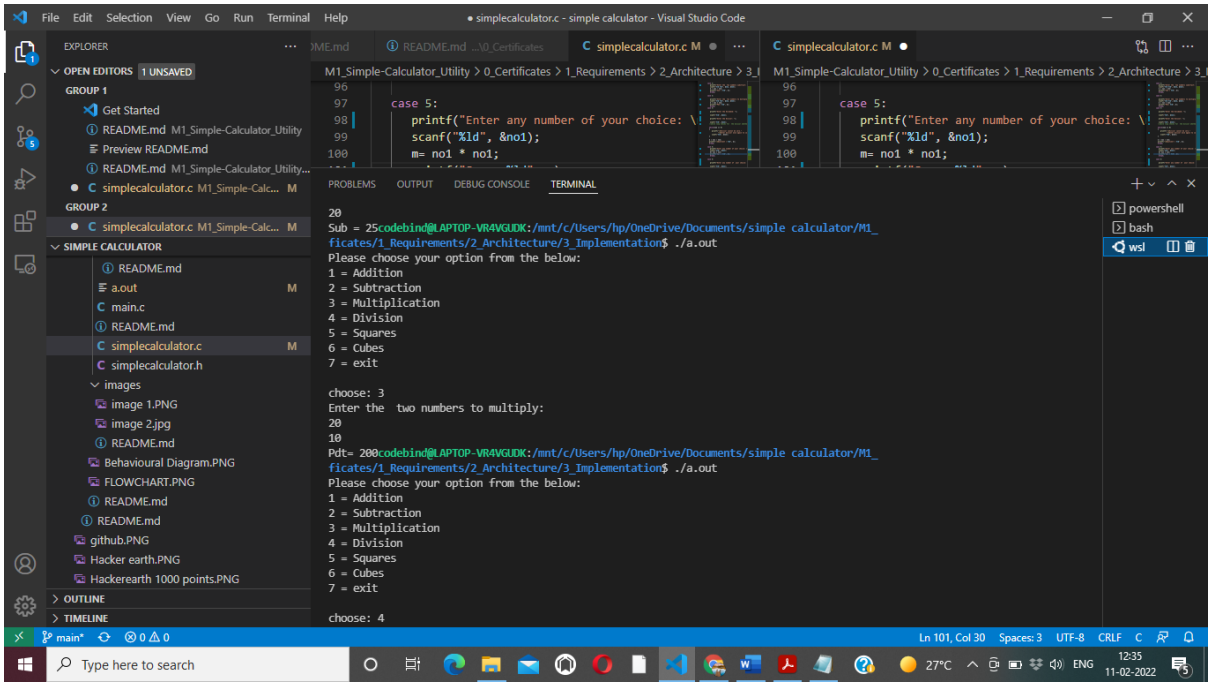
```
{
printf("\nDivisor cannot be zero."
"\nEnter divisor once again to confirm: ");
scanf("%ld", &no2);
}

m = no1 / no2;
printf("\nQuotient = %ld", m);
break;

case 5:
printf("Please enter any number of your choice: \n");
scanf("%ld", &no1);
m= no1 * no1;
printf("Sqr = %ld", m);
break;

case 6:
printf("Please enter any number of  your choice: \n");
scanf("%ld", &no1);
m = no1 * no1 * no1;
printf("Cube = %ld", m);
break;
case 7:
return 0;
```

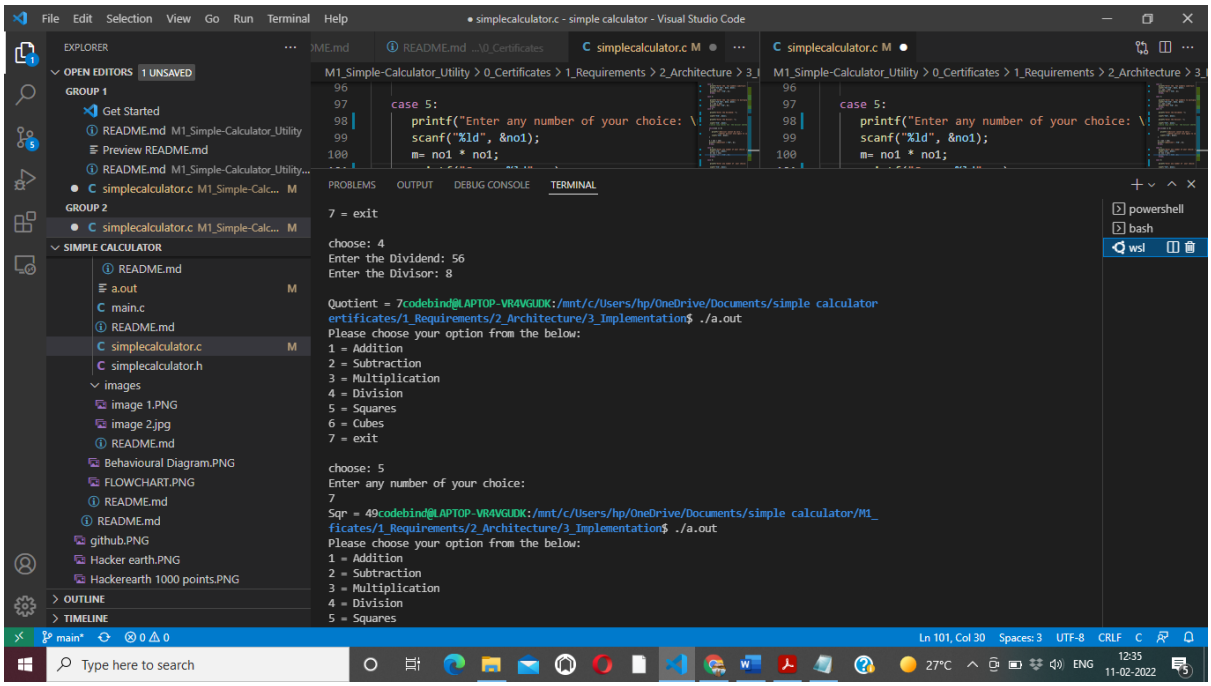
```
default: printf("\nError");
}
}
```

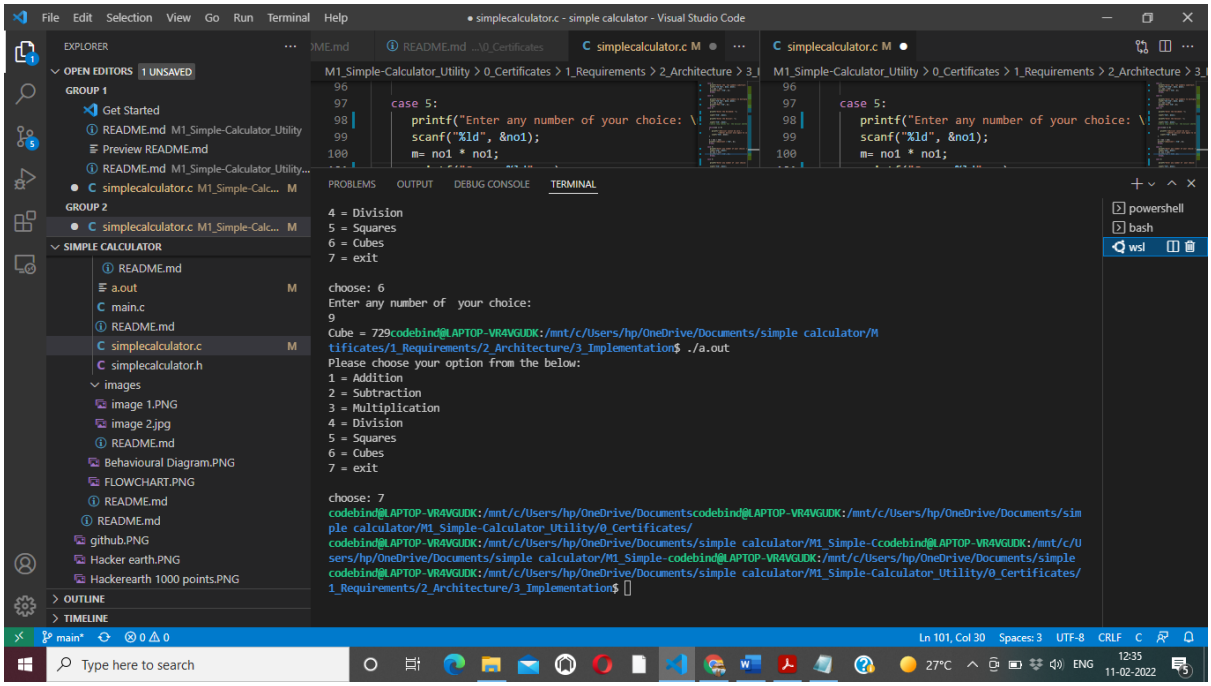
❖ OUTPUT(includes all the results of various operations)






File Edit Selection View Go Run Terminal Help
• simplecalculator.c - simple calculator - Visual Studio Code
EXPLORER
OPEN EDITORS 1 UNSAVED
GROUP 1
Get Started
README.md M1_Simple-Calculator_Utility
Preview README.md
README.md M1_Simple-Calculator_Utility...
simplecalculator.c M1_Simple-Calc... M
GROUP 2
simplecalculator.c M1_Simple-Calc... M
SIMPLE CALCULATOR
README.md
a.out M
main.c
README.md
simplecalculator.c M
simplecalculator.h
images
image 1.PNG
image 2.jpg
README.md
Behavioural Diagram.PNG
FLOWCHART.PNG
README.md
README.md
github.PNG
Hacker earth.PNG
Hackerearth 1000 points.PNG
OUTLINE
TIMELINE
simplecalculator.c M
M1_Simple-Calculator_Utility > 0_Certificates > 1_Requirements > 2_Architecture > 3_I
case 5:
printf("Enter any number of your choice: \
scanf("%ld", &no1);
m= no1 * no1;
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
powershell
bash
wsl
20
Sub = 25codebind@LAPTOP-VR4VGUDK:/mnt/c/Users/hp/OneDrive/Documents/simple calculator/M1_
ficates/1_Requirements/2_Architecture/3_Implementation$ ./a.out
Please choose your option from the below:
1 = Addition
2 = Subtraction
3 = Multiplication
4 = Division
5 = Squares
6 = Cubes
7 = exit
choose: 3
Enter the two numbers to multiply:
20
10
Pdt= 200codebind@LAPTOP-VR4VGUDK:/mnt/c/Users/hp/OneDrive/Documents/simple calculator/M1_
ficates/1_Requirements/2_Architecture/3_Implementation$ ./a.out
Please choose your option from the below:
1 = Addition
2 = Subtraction
3 = Multiplication
4 = Division
5 = Squares
6 = Cubes
7 = exit
choose: 4
main*
Ln 101, Col 30 Spaces: 3 UTF-8 CRLF C
Type here to search
27°C
ENG
12:35
11-02-2022


File Edit Selection View Go Run Terminal Help
• simplecalculator.c - simple calculator - Visual Studio Code
EXPLORER
OPEN EDITORS 1 UNSAVED
GROUP 1
Get Started
README.md M1_Simple-Calculator_Utility
Preview README.md
README.md M1_Simple-Calculator_Utility...
simplecalculator.c M1_Simple-Calc... M
GROUP 2
simplecalculator.c M1_Simple-Calc... M
SIMPLE CALCULATOR
README.md
a.out M
main.c
README.md
simplecalculator.c M
simplecalculator.h
images
image 1.PNG
image 2.jpg
README.md
Behavioural Diagram.PNG
FLOWCHART.PNG
README.md
README.md
github.PNG
Hacker earth.PNG
Hackerearth 1000 points.PNG
OUTLINE
TIMELINE
simplecalculator.c M
M1_Simple-Calculator_Utility > 0_Certificates > 1_Requirements > 2_Architecture > 3_I
case 5:
printf("Enter any number of your choice: \
scanf("%ld", &no1);
m= no1 * no1;
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
powershell
bash
wsl
7 = exit
choose: 4
Enter the Dividend: 56
Enter the Divisor: 8
Quotient = 7codebind@LAPTOP-VR4VGUDK:/mnt/c/Users/hp/OneDrive/Documents/simple calculator
ertificates/1_Requirements/2_Architecture/3_Implementation$ ./a.out
Please choose your option from the below:
1 = Addition
2 = Subtraction
3 = Multiplication
4 = Division
5 = Squares
6 = Cubes
7 = exit
choose: 5
Enter any number of your choice:
7
Sqr = 49codebind@LAPTOP-VR4VGUDK:/mnt/c/Users/hp/OneDrive/Documents/simple calculator/M1_
ficates/1_Requirements/2_Architecture/3_Implementation$ ./a.out
Please choose your option from the below:
1 = Addition
2 = Subtraction
3 = Multiplication
4 = Division
5 = Squares
main*
Ln 101, Col 30 Spaces: 3 UTF-8 CRLF C
Type here to search
27°C
ENG
12:35
11-02-2022



5.Test plan

1. For the Addition operation after taking two numbers we use '+' operand to do Addition .
2. For the Substraction operation after taking two numbers we use '-' operand to do Substraction.
3. For the Multiplication operation after taking two numbers we use '*' operand to do Multiplication.
4. For the Division operation after taking two numbers we use '/' operand to do Division .
5. For the Squaring operation after taking one number we use '*' operand to get the square of the particular number .
6. To Cube a particular number we use '*' opearnd twice in the operation.

6.REFERENCES:

- Referred some raw ideas from web platform and while analysing the code for the project.