
# Lab Notebook

Software Tools and Technology Lab

SEBCA1191

| ***Group 19*** | | |
|---|---|---|
| NAME | ROLL NO. | DEPARTMENT |
| Rishika Acharjee [Group-Leader] | 30001223016 | BCA |
| Ankita Ghoshal | 330085323012 | Bsc IT in CS |
| Moumita Modak | 30054623014 | Bsc IT in AI |
| Pritam Sarkar | 30001223042 | BCA |
| Titli Biswas | 30059223018 | BSc in Data Science |

# Index

# Member 2: [Ankita Ghoshal]

## 1 Lab 1: Calculator Program using C

### 1.1 Objective

The objective of this lab is to develop a basic calculator program using the C programming language. The calculator will perform simple arithmetic operations like addition, subtraction, multiplication, and division based on user input.

### 1.2 Program Overview

The calculator program is designed to:

- Accept two numbers from the user.
- Prompt the user to select an arithmetic operation (Addition, Subtraction, Multiplication, Division).
- Perform the selected operation.
- Display the result of the operation to the user.

The program includes error handling to manage division by zero and other invalid inputs.

### 1.3 Code Implementation

The following is the C code for the calculator program:

```
#include <stdio.h>

int main() {
    char operator;
    double num1, num2, result;

    printf("Enter an operator (+, -, *, /): ");
    scanf("%c", &operator);

    printf("Enter two operands: ");
    scanf("%lf %lf", &num1, &num2);

    switch(operator) {
        case '+':
            result = num1 + num2;
            break;
```

```
        case '-':
            result = num1 - num2;
            break;
        case '*':
            result = num1 * num2;
            break;
        case '/':
            if (num2 != 0)
                result = num1 / num2;
            else {
                printf("Error! Division by zero.\n");
                return -1;
            }
            break;
        default:
            printf("Error! Operator is not correct\n");
            return -1;
    }

    printf("Result: %.2lf\n", result);
    return 0;
}
```

## 1.4 Compiling and Running the Program

To compile and run the calculator program:

1. Open a terminal or command prompt.
2. Navigate to the directory where the C file is located.
3. Compile the program using a C compiler (e.g., GCC):

```
gcc calculator.c -o calculator
```

4. Run the compiled program:

```
./calculator
```

## 1.5 Adding the Calculator Program to GitHub Repository

To add this calculator program to a GitHub repository, follow these steps:

#### 1.5.1 Step 1: Initialize a Local Git Repository

1. Open the terminal and navigate to the directory where your `calculator.c` file is located.
2. If you haven't already, initialize a Git repository in that directory:

   ```
   git init
   ```

   This command creates a new Git repository in the current directory.

#### 1.5.2 Step 2: Add the File to the Repository

1. Add the `calculator.c` file to the staging area:

   ```
   git add calculator.c
   ```

   This command stages the file, indicating that you want to include it in the next commit.

#### 1.5.3 Step 3: Commit the Changes

1. Commit the file to the repository with a meaningful message:

   ```
   git commit -m "Add calculator program in C"
   ```

#### 1.5.4 Step 4: Push the Changes to GitHub

1. Link your local repository to a remote GitHub repository:

   ```
   git remote add origin https://github.com/yourusername/your-repo-name.git
   ```

2. Push the changes to the GitHub repository:

   ```
   git push -u origin master
   ```

#### 1.5.5 Step 5: Verify the Upload

1. Go to your GitHub repository URL in a web browser.
2. Verify that the `calculator.c` file is listed and accessible in the repository.

# Member 4: [Pritam Sarkar]

## 2 Introduction

This document outlines the process of modifying a "Submit" button in a mind reader application and submitting a pull request to the original GitHub repository. The repository in question is available at `https://github.com/GeekAyan/STT`. The modification includes renaming the button and fixing proportion issues.

### 2.1 Cloning the GitHub Repository

**Step**: Clone the GitHub repository using GitHub Desktop.

**Action**:

- Open GitHub Desktop and select `File > Clone Repository`.
- Enter the repository URL: `https://github.com/GeekAyan/STT` and select a directory to clone it.

### 2.2 Opening the Project in an IDE

**Step**: Open the cloned project using your preferred IDE (e.g., VS Code, PyCharm).

**Action**:

- Open the folder containing the cloned project.
- Review the `README.md` for instructions on how to run the project.

### 2.3 Install Dependencies

**Step**: Install any dependencies required by the project as per the `README.md` file.

**Action**:

- Set up the environment. If the project uses Python, create a virtual environment and install dependencies using:

```
pip install -r requirements.txt
```

- Follow other system requirements mentioned in the `README.md`.

### 2.4 Running the Application

**Step**: Run the application as per the instructions in `README.md`.

**Action**:

- Use your IDE's terminal to run the project.
- Ensure the application works as expected.

## 2.5 Renaming the Submit Button

**Step**: Rename the button from "Submit" to "Chin Tapak Dum Dum."
**Action**:

- Find the code section responsible for the submit button's label.
- Modify the label. For example:

```
<button id="submit" name="submit">Chin Tapak
    Dum Dum</button>
```

## 2.6 Fixing the Button Proportion

**Step**: After renaming the button, analyze and adjust its proportions.
**Action**:

- Check the CSS properties related to the button's size, padding, and font.
- Modify the button's CSS if needed, for example:
- Save changes and re-run the application to check the button's appearance.

## 2.7 Testing the Application

**Step**: Test the application after modifying the button.
**Action**:

- Run the application again to verify that the button looks correct and functions properly.

## 2.8 Committing the Changes

**Step**: Commit your changes locally.
**Action**:

- Stage the files and commit with a descriptive message, for example:

```
git commit -m "Renamed␣submit␣button␣and␣fixed
    ␣proportion␣issue"
```

## 2.9 Pushing Changes to Your Fork

**Step**: Push your changes to your GitHub fork.
**Action**:

- If you haven't forked the repository, go to the GitHub page and fork it.
- Add the forked repository as a remote and push your changes:

```
git remote add origin https://github.com/<
    YourGitHubUsername>/STT.git
git push origin main
```

## 2.10 Creating a Pull Request

**Step**: Create a pull request to the original repository.
**Action**:

- Go to your fork on GitHub and create a new pull request.
- Provide a descriptive title and details of your changes.
- Submit the pull request.

## 2.11 Review and Approval

**Step**: Wait for feedback or approval. **Action**:

- Respond to feedback or requested changes, if any.
- Once approved, your changes will be merged into the original project.

# Latex Assignment by Pabitra Sir

## Entry by Titli Biswas

## 2.12 Mathematical Notation

1. Superscripts, subscripts, and Greek letters
   - 2224
   - $22_{24}$
   - $2224^{\pi}$
   - $\cos\theta$
   - $\tan^{-1}(2.224)$
   - $\log_{22} 24$
   - $\ln 2224$
   - $e^{2.224}$
   - $0 < x \leq 2224$
   - $y \geq 2224$
2. Roots, fractions, and displaystyle
   - $\sqrt{2224}$
   - $\sqrt[22]{24}$
   - Normal: $\frac{22}{24}$ Displaystyle: $\displaystyle\frac{22}{24}$
   - Normal: $\frac{2}{2+\frac{2}{4}}$ Displaystyle: $\displaystyle\frac{2}{2+\frac{2}{4}}$
   - Normal: $\sqrt{\frac{22}{24}}$ Displaystyle: $\displaystyle\sqrt{\frac{22}{24}}$
3. Delimiters
   - Display math mode: $\displaystyle\left(2+\frac{2}{24}\right)$
   - Display math mode: $\displaystyle\left|\frac{22-2}{4}\right|$
4. Tables and equation arrays
   -

| $x$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $f(x)$ | 2 | 2 | 2 | 2 |

- $2 + 2 - 2 \times 4 = x$
  $2 + 2 - 8 = x$
  $4 - 6 = x$
  $x = -2$

5. Functions & Formulas

   - The quadratic formula:

   $$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

   - The function $f(x) = \left(\frac{x+2}{2}\right)^2 - \frac{2}{4}$ has domain $D_f : (-\infty, \infty)$ and range $R_f : \left[\frac{-7}{2}, \infty\right)$.
   - Definition of a Derivative: $\lim_{h \to 0} \frac{f(x+h) - f(x)}{h} = f'(x)$
   - Chain Rule: $[f(g(x))]' = f'(g(x)) \cdot g'(x)$
   - $\frac{d^2y}{dx^2} = f''(x)$
   - $\int \sec^2 x \, dx = \tan x + C$
   - $\int e^{2x} \, dx = \frac{1}{2} e^{2x} + C$
   - Fundamental Theorem of Calculus, Part 1: $\int_a^b f'(x) \, dx = f(b) - f(a)$
   - Fundamental Theorem of Calculus, Part 2: $\frac{d}{dx}\left(\int_a^{g(x)} f(t) \, dt\right) = f(g(x)) \cdot g'(x)$
   - Euler's Method: $y_1 = y_0 + hF(x_0, y_0)$ where $h$ is the step size, and $F(x, y) = \frac{dy}{dx}$
   - $a_n = \{2224, \frac{2224}{2}, \frac{2224}{4}, \frac{2224}{8}, \cdots, \frac{2224}{2^n}\}$ represents a geometric sequence.
   - $S_n = \sum_{n=1}^{\infty} \frac{2224}{2^n}$ is a convergent geometric series since $|r| = \left|\frac{1}{2}\right| < 1$.
   - Taylor Series: $\sum_{n=0}^{\infty} \frac{f^{(n)}(c)}{n!}(x - c)^n$
   - Velocity Vector: $\vec{v}(t) = x'(t)\vec{i} + y'(t)\vec{j} = \left\langle \frac{dx}{dt}, \frac{dy}{dt} \right\rangle$
   - Area of Polar Curve: $A = \frac{1}{2} \int_\alpha^\beta r^2 \, d\theta$