

MAULANA ABUL KALAM AZAD  
UNIVERSITY OF TECHNOLOGY,  
WEST BENGAL



# Lab Notebook

Software Tools and Technology Lab

SEBCA1191

<i><b>Group 19</b></i>		
NAME	ROLL NO.	DEPARTMENT
Rishika Acharjee [Group-Leader]	30001223016	BCA
Ankita Ghoshal	330085323012	Bsc IT in CS
Pritam Sarkar	30001223042	BCA
Titli Biswas	30059223018	BSc in Forensic Science
Moumita Modak	30054623014	Bsc IT in AI

# Index

Sl. No.	Experiment	Page No.
1	Introduction to Latex	1
2	Calculator Program using C	3
3	Converting a submit button to Chin Tapak Dum Dum	7
4	Mathematical Notation	10
5	CV	12

# Acknowledgment

We gratefully acknowledge the efforts of all group members in contributing their respective lab notebook entries and committing them to the GitHub repository. We extend our sincere gratitude to the subject-faculties for their valuable guidance and support throughout this assignment. This experience has significantly enhanced our skills in LaTeX and collaborative work.

---

Student's Signature

# Group-Leader: [Rishika Acharjee]

## 1 Introduction to L<sup>A</sup>T<sub>E</sub>X

is a typesetting system that is widely used for producing scientific and technical documents. It is especially popular for creating documents with complex mathematical formulas, tables, and figures. Unlike word processors, L<sup>A</sup>T<sub>E</sub>X provides greater control over document structure and presentation, making it a preferred tool in academia and industry.

### 1.1 Key Features of L<sup>A</sup>T<sub>E</sub>X

Some of the key features of L<sup>A</sup>T<sub>E</sub>X include:

- High-quality typesetting, especially for mathematical and technical content.
- Separation of content and style, allowing users to focus on writing.
- Automated table of contents, bibliography, and cross-references.
- Supports a wide range of document types, including articles, books, reports, and presentations.
- Extensible through packages for additional functionality, such as graphics, tables, and advanced formatting.

### 1.2 Why Use L<sup>A</sup>T<sub>E</sub>X?

There are several reasons to use L<sup>A</sup>T<sub>E</sub>X:

- **Precision:** L<sup>A</sup>T<sub>E</sub>X provides exceptional control over document structure and formatting, ensuring that everything looks just right.
- **Mathematical Formulas:** It is the standard for creating documents that contain mathematical symbols and equations.
- **Professional Quality:** Documents created with L<sup>A</sup>T<sub>E</sub>X have a professional appearance, suitable for academic papers, theses, and books.
- **Consistency:** Once a style is defined, it is applied consistently throughout the document.
- **Collaboration:** Multiple users can work on the same document without formatting issues, as L<sup>A</sup>T<sub>E</sub>X is plain text-based.

### 1.3 Creating a L<sup>A</sup>T<sub>E</sub>X Repository on GitHub using GitHub Desktop

Version control is crucial for managing changes in documents, especially when working on complex projects. GitHub provides a platform for managing repositories, and GitHub Desktop simplifies working with GitHub.

To create a L<sup>A</sup>T<sub>E</sub>X repository on GitHub using GitHub Desktop:

1. **Install GitHub Desktop:** Download and install GitHub Desktop from <https://desktop.github.com/>.
2. **Create a GitHub Account:** If you don't have one, sign up for GitHub at <https://github.com/>.
3. **Create a New Repository:** Open GitHub Desktop and click on *File* → *New Repository*. Name your repository, select the local path, and ensure the repository is initialized with a README.
4. **Clone the Repository:** After creating the repository on GitHub, use GitHub Desktop to clone the repository to your local machine by clicking on *File* → *Clone Repository*.
5. **Add Your L<sup>A</sup>T<sub>E</sub>X Files:** Open the cloned folder on your computer and add your L<sup>A</sup>T<sub>E</sub>X source files (.tex, .bib, etc.).
6. **Commit and Push:** After adding your files, return to GitHub Desktop, write a commit message, and click on *Commit to main*. Then click on *Push origin* to upload your changes to GitHub.

A large, stylized logo for L<sup>A</sup>T<sub>E</sub>X, rendered in a black serif font. The letters are large and closely spaced, with the 'A' having a superscript 'L' and the 'E' having a subscript 'X'.

Figure 1: L<sup>A</sup>T<sub>E</sub>X

## Member 2: [Ankita Ghoshal]

### 2 Calculator Program using C

#### 2.1 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.

#### 2.2 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.

#### 2.3 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;
}

```

## 2.4 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
```

## 2.5 Adding the Calculator Program to GitHub Repository

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

### 2.5.1 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.

### 2.5.2 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.

### 2.5.3 Step 3: Commit the Changes

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

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

### 2.5.4 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
```

### 2.5.5 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.

The given code takes two double inputs (`num1` and `num2`), performs a mathematical operation based on the operator provided, and prints the result. Below are the possible outputs depending on the operator and inputs.

## 2.6 1. Addition Case ('+' Operator)

If the operator is '+', the code will add `num1` and `num2` and output the result.

Input: 5.2, 3.8, +

Output: Result: 9.00



## 2.7 2. Subtraction Case ('-' Operator)

If the operator is '-', the code will subtract `num2` from `num1` and output the result.

Input: 10.5, 4.2, -

Output: Result: 6.30

## 2.8 3. Multiplication Case ('\*' Operator)

If the operator is '\*', the code will multiply `num1` and `num2` and output the result.

Input: 7.0, 3.0, \*

Output: Result: 21.00

## 2.9 4. Division Case ('/' Operator)

If the operator is '/' and `num2` is not zero, the code will divide `num1` by `num2` and output the result.

Input: 20.0, 4.0, /

Output: Result: 5.00

## 2.10 5. Division by Zero Case

If the operator is '/' and `num2` is zero, the code will print an error message and return -1.

Input: 10.0, 0, /

Output: Error! Division by zero.

## 2.11 6. Invalid Operator Case

If an invalid operator is provided, the code will print an error message and return -1.

Input: 5.0, 3.0, ^

Output: Error! Operator is not correct.

## Member 3: [Pritam Sarkar]

### 3 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.

#### 3.1 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.

#### 3.2 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.

#### 3.3 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`.

#### 3.4 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.

### 3.5 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>
```

### 3.6 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.

### 3.7 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.

### 3.8 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"
```

### 3.9 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
```

### 3.10 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.

### 3.11 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.

## Member 4: TITLI BISWAS

### 4 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:  $\frac{22}{24}$
- Normal:  $\frac{2}{2+\frac{2}{4}}$  Displaystyle:  $\frac{2}{2+\frac{2}{4}}$
- Normal:  $\sqrt{\frac{22}{24}}$  Displaystyle:  $\sqrt{\frac{22}{24}}$

#### 3. Delimiters

- Display math mode:  $\left(2 + \frac{2}{24}\right)$
- Display math mode:  $\left|\frac{22-2}{4}\right|$

index

#### 4. Tables and Equation Arrays

(a)

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

(b)

$$1 + 1 - 2 \times 3 = x \quad (1)$$

$$1 + 1 - 6 = x \quad (2)$$

$$2 - 6 = x \quad (3)$$

$$x = -4 \quad (4)$$

## 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:  $\lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} = f'(x)$
- 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  $F(x, y) = \frac{dy}{dx}$
- $a_n = \{2224, \frac{2224}{2}, \frac{2224}{4}, \frac{2224}{8}, \dots, \frac{2224}{2^n}\}$  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:  $A = \frac{1}{2} \int_{\alpha}^{\beta} r^2 \, d\theta$

---

Member 5:Moumita Modak

# CV

## Moumita Modak

✉ modakmoumita789@gmail.com    ☎ +91-8335920022  
🌐 github.com/moumitamodak    in linkedin.com/in/moumita-modak-1797062b9

### Education

---

**BSc in IT (Artificial Intelligence)**, Maulana Abul Kalam Azad **2023 – 2027**  
University Of Technology, Kalyani

GPA: 8.0/10.0

### Skills

---

<b>Programming Languages:</b>	HTML, CSS, JavaScript, C, C++, Python
<b>Web Development:</b>	React, Bootstrap, jQuery
<b>Tools:</b>	Git, GitHub, VS Code, Postman
<b>Design:</b>	Photoshop, Illustrator

### Experience

---

#### Web Development Projects

---

- Developed responsive websites using HTML, CSS, JavaScript.
- Worked with peers on user-friendly interfaces.
- Integrated RESTful APIs in projects.

#### Project: SymbolApp

---

- Created a Java AWT application to manage symbols.
  - Implemented features like creation, deletion, modification.
-

---

## Projects

---

### LaTeX Lab Notebook

---

- Created a lab notebook using LaTeX with indexing and custom formatting.

### Music Player

---

- Developing a JavaScript music player with playlist management.

## Interests

---

- Gaming robotics,military applications,expert systems,search engines.
-