

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 1 of 13

CMPS1134 – Fundamentals of Computing**Final Project – Binary calculator Application (task 1)****Due Date:** Wednesday, 26 November 2025

SIGNATURE BLOCK				
Statement	I did my share of the work, and I have a general understanding of the contents of the assignment.			
Team Member	Contribution	% of Total	Signature	Data
Jenner Velasquez	Planning - Developed the project plan and organized the workflow.	25%		
Kevin Taca	Design – created the webpage layout and user interface.	25%		
Nikita Link	Coding – implemented the website's features and functionality.	25%		
Macario Cus	Calculator- Developed the logic and operational features of the binary calculator.	25%		.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 2 of 13

Table of Contents

1.0 Introduction	-----
1.1 Course Overview	-----
1.2 Project objective	-----
1.3 Project Description	-----
1.4 Importance of the Project in Computing	-----
2.0 The Program	-----
2.1 Work Distribution Among Team Members	-----
2.2 Task Breakdown and Responsibilities	-----
2.3 Tools, Technologies, and Resources Used	-----
2.4 Implementation Process	-----
2.5 Challenges Faced and Solutions	-----
3.0 Lecture Connection	-----
3.1 Relevant Lecture Topic	-----
3.2 Concepts Applied from Lecture	-----
3.3 Examples of Application in Code	-----
4.0 Code and Demonstration	-----
4.1 Overview of Code Structure	-----
4.2 Sample Code Snippets and Explanations	-----
4.3 Testing and Debugging Process	-----
5.0 Conclusion	-----
References	-----
Appendices	-----

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 3 of 13

1.0 Introduction

1.1 Course Overview

Fundamentals of Computing(CMPS1134) introduces students to the main concepts of computer science. Students learn how computers store and process information, how programming logic is used to break problems into smaller steps, and how data is organized so that it can be applied in real solutions. These topics are explored through lectures, discussions, class assessments, and a final project. By the end of the semester, students are able to apply theoretical concepts learned to topics learned in the real world.

1.2 Project Objective

The objective of the project was for students to take the theoretical concepts learned in class and apply them in a practical way by building a functioning binary arithmetic calculator. This allowed students to use programming logic, binary representation, and design a simple, user-friendly interface, and to develop communication, collaboration, UI design, and debugging skills.

1.3 Project Description

The project included creating a homepage, an introduction page, a projects page, a documentation page, and a binary calculator page. The Binary Calculator allows users to convert numbers between binary and decimal and perform basic binary operations, addition and subtraction. The overall design of the websites was kept simple so users would be able to easily move between pages and understand how to use the calculator.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 4 of 13

1.4 Importance of the Project in Computing

The project is important because it allows students to use the important concepts from the course in a practical assignment. Building the website pages in groups strengthened students' understanding of HTML, CSS, and JavaScript, and developed their problem-solving and debugging skills.

This project was valuable because it took the ideas learned in class and placed them in a real situation where we had to build something that worked.

2.0 The Program

2.1 Work Distribution Among Team Members

The team members responsible for the Binary Calculator were Kevin Taca, Nikita Link, Macario Cus, and Jenner Velasquez. Kevin divided the work evenly based on each person's strengths. He created the project outline, organized the workflow, made sure everyone followed the rubric, and wrote section four of the report. Macario handled the design layout of the website. He built the HTML structure, set up the buttons, input fields, and display areas for the calculator. He also styled the pages using CSS with a focus on keeping the layout clean, readable, and consistent across all pages. He also wrote section two. Nikita worked on the JavaScript. She wrote the logic for the addition and subtraction functions and made sure the calculator responded correctly to user inputs. She also kept the site design simple so that users could move through the pages easily. Additionally, Nikita wrote sections one and three of the report.

Jenner was responsible for testing and quality control. He checked every feature of the website to make sure it functioned correctly, including testing invalid inputs, edge cases, and long binary values. He also reported bugs, suggested fixes, and confirmed that the layout displayed correctly on different devices. Additionally, he

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 5 of 13

helped format the final report, so the document was well-organized and professional. Jenner also wrote section five of the report.

2.2 Task Breakdown and Responsibilities

The project was divided evenly among the four team members. Planning and Coordination tasks required team members to understand the project's requirements and decide which pages the website should include. Team members had to agree on the features of the calculator and discuss which tools should be used and for what purpose. They also had to set deadlines and create a simple plan so each person knew what to focus on. Website Design and Layout: The team had to plan the overall theme and look of the website by choosing the colour theme, spacing, fonts, and layout. They wanted the site to be clean and easy to navigate, so they worked on organizing the elements clearly and making sure the design works on both large and small screens. HTML Structure and Page Setup: The webpages were built separately. The webpage included adding headings, text sections, navigation links, buttons, and containers for information. The team made sure that all pages were linked properly so users would be able to move around easily. JavaScript Development: For the calculator to work, the team had to write functions that handled binary and decimal conversions, validated the input, and performed arithmetic operations. Because JavaScript struggles with very large numbers, BigInt was used to prevent errors from occurring. Testing and Debugging: The calculator was tested using valid and invalid inputs, including long binary numbers and empty fields. After errors appeared, the console and developer tools were used to fix the problems. Documentation and Final Report: This involved writing explanations, collecting screenshots, adding sample code and preparing the final version of the report. The goal was to clearly show how the calculator and website were built and to present the project in a professional manner.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 6 of 13

2.3 Tools, Technologies, and Resources Used

Tools, Technologies, and Resources	Use
HTML, CSS, JavaScript	Forming the foundation of the website
Visual Studio Code	Used for writing and organizing code
Browser Developer Tools	Helped in testing and debugging the layout and functions
Google Drive	Sharing videos, files, screenshots, and report drafts
GitHub and GitHub Pages	Storing the code and publishing the website online
WhatsApp	Communication
ChatGPT	Helping in refining the design template and solving small coding issues

2.4 Implementation Process

The project was started by creating the main index.html file since it served as the Home Page and the base for all the pages. After creating that page, we created the introduction, Projects, Binary Calculator, and Documentation pages. The pages were linked through the navigation bar and checked to ensure that each page link worked. Then, the main focus was the design. The colours, spacing, and layouts were adjusted until the pages looked clean and consistent. The CSS made the site responsive so it would work on both phones and computers. After the layout was complete. JavaScript was used to write functions for converting values and performing operations. Connecting these functions to the buttons took time, since every small error in the code affected the results and overall

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 7 of 13

functionality of the website. wherever issues were found and went beyond our mental capacity, ChatGPT was implemented. Lastly, the website was tested thoroughly across all platforms, using different inputs. The calculator response was observed, and developer tools were used to find and fix problems. Once everything was functioning properly, the project was uploaded to GitHub to publish.

2.5 Challenges Faced and Solutions

Challenges	Solutions
Inconsistent design across some pages	Creating a shared CSS stylesheet kept the layout consistent across all pages, and ChatGPT showed us how to apply the CSS stylesheet.
Calculator errors with large binary numbers	Using JavaScript's BigInt feature with the aid of ChatGPT.
Hard to find bugs in the layout and code functions	Used browser developer tools to inspect elements and test.

3.0 Lecture Connection

3.1 Concepts Applied from Lecture

The lecture that connected most directly to this project was the chapter on Data Manipulation. This section explained how computers represent information in binary and how the CPU processes instructions through the fetch–decode–execute cycle. It also covered how the Arithmetic Logic Unit (ALU) performs basic operations such as addition and subtraction. These concepts were important for our project because the binary calculator depends on binary representation, input validation, and arithmetic operations that mirror the ALU's functions

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 8 of 13

3.2 Concepts Applied from Lecture

While the team drew on several parts of the course, the concept that showed up most often during development was the fetch–decode–execute cycle. In our calculator, the steps matched the cycle almost exactly. When a user enters a number, the value is first fetched from the text box. The code then checks whether the input contains only 0s and 1s, which works like the decode stage, where the CPU examines whether an instruction is valid. If the input is acceptable, the calculator performs the required operation and displays the answer on the screen; this is the execute stage. Seeing these steps in our own code helped us understand how the cycle works in practice instead of only reading about it in the chapter.

3.3 Example from Code

```
function addBinary() {  
  const a = g("binA").value.trim(), b = g("binB").value.trim();  
  if (!/^[01]+$/.test(a) || !/^[01]+$/.test(b)) {  
    alert("Enter valid binary numbers (0 and 1 only)");  
    return;  
  }  
  const result = (BigInt("0b" + a) + BigInt("0b" + b)).toString(2);  
  g("op-result").textContent = result  
}
```

4.0 Code and Demonstration

4.1 Overview of Code Structure

The website was organized into separate files to keep everything organized: HTML files were used for the structure of each page, style.css was used for designing and layout, and script.js was used for calculating logic and validation functions. Separating files made the website easier to manage and update.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 9 of 13

4.2 Sample code

```
function sanitizeBinary(el) {  
  if (!el) return;  
  el.value = el.value.replace(/[^01-]/g, '').replace(/-+/g, '-')  
}
```

This code checks whether the user entered a valid binary number. It removes anything that is not a 0 or 1. The simple function works like the “decode” step in the Fetch Decode Execute cycle. When the user types something in the textbox, the function immediately checks the characters and removes anything that is not binary. This makes sure the calculator only works with proper binary values, just like the CPU only works with valid instructions.

4.3 Testing and Debugging Process

To make sure the calculator worked properly, each part of the project was tested as soon as it was added. We started by checking simple things, such as whether the input boxes accepted only valid numbers and whether the buttons responded when clicked. After that, we tested each function with different types of inputs, including normal cases, large numbers, and incorrect values like letters or blank spaces.

When something did not work as expected, we used the browser’s developer tools to inspect the code, check for errors, and track where the problem started. Most of the bugs originated from typing mistakes, missing brackets, or functions not being properly linked to the HTML buttons. We fixed these issues one by one and re-tested the calculator to make sure the changes did not cause new problems.

We also tested the layout on both phones and computers to make sure the design stayed consistent and readable. By the end of the process, the calculator was able to handle conversions and binary operations smoothly, and all the major errors had been corrected.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 10 of 13

5.0 Conclusion

Working on this project helped our team bring together the main ideas from the course in a practical way. Building the binary calculator showed us how concepts like data representation, binary arithmetic, and basic programming logic work when applied to a real task. Each stage, from planning the layout to writing JavaScript and testing the calculator, required us to communicate, solve problems, and depend on one another's strengths.

The project also gave us a clearer understanding of how HTML, CSS, and JavaScript connect to create a working webpage. Seeing the calculator function after fixing errors and improving the design made the entire process feel meaningful. Overall, this assignment strengthened our technical skills and our ability to work as a team, and it prepared us for more advanced computing tasks in the future.

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 11 of 13

References:

Data Manipulation 1:

https://doit.ub.edu.bz/pluginfile.php/28297/mod_resource/content/7/Lectures/CMPS1134-LC02-DataManipulation-1-20S2.pdf

Data Manipulation 2:

https://doit.ub.edu.bz/pluginfile.php/28298/mod_resource/content/9/Lectures/CMPS1134-LC02-DataManipulation-2-18S1.pdf

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 12 of 13

Appendix A: Screenshot of the binary calculator webpage interface

UB Logo
University Of Belize
Fundamentals Of Computing · CMPS1134

Home Introduction Projects **Binary Calculator** Documentation

Binary — Quick Overview

Binary is a base-2 number system that uses only two digits, 0 and 1, to represent all information.

How binary works

- **Base-2 system:** Unlike decimal (base-10), binary has two digits: 0 and 1.
- **On/Off states:** Electronic circuits use 0 and 1 to represent off/on.
- **Bits & bytes:** A single binary digit is a bit. Eight bits = 1 byte.

Example: 2 in binary is 10. 3 is 11.

Convert between Binary & Decimal

Binary (e.g., 1010) Decimal (e.g., 42)

Binary → Decimal **Decimal → Binary** Swap Clear

Result Type: Result:

Binary Addition & Subtraction

1st Binary (e.g., 1010) 2nd Binary (e.g., 1101)

Add **Subtract** Clear

Operation Result:

Assignment Number	01
Version	01
Print Date	17/11/25
Page	Page 13 of 13

Appendix B: AI Declaration Form

Declaration of Use for Generative AI in Assessments



I hereby declare that in the planning, drafting, and/or revision of the work attached, I have made use of generative AI tools in the following ways:

Acknowledgement of Generative AI Tools Used

- | | |
|--|---|
| <input checked="" type="checkbox"/> For brainstorming | <input type="checkbox"/> To generate translations included in the submitted work, whether or not manually revised |
| <input type="checkbox"/> To find sources | <input checked="" type="checkbox"/> To improve the language of my own phrases, sentences, and/or paragraphs. |
| <input checked="" type="checkbox"/> To plan the structure/outline of the work | <input type="checkbox"/> To generate the text of (part of) the submitted work |
| <input checked="" type="checkbox"/> To generate programming code | |
| <input type="checkbox"/> To generate translations of primary/secondary source content for consultation | |

Acknowledgement of Assessment Submission

I, **Macario Cus**, hereby confirm that on **11 November 2025**:

1. I am the author of this submitted document.
2. I am responsible for any AI-generated errors or fabrications.
3. I understand the limitations and risks of using AI.
4. I used AI tools ethically, protecting all sensitive information.
5. I ensure any AI-assisted work remains originally my own.
6. I have appropriately acknowledged all use of generative AI.
7. Undeclared AI use constitutes academic dishonesty, which I acknowledge.
8. I am accountable for any resulting academic misconduct.

Please add more rows to the table as needed so that each tool used in the creation of your assessment submission are included in the declaration.


Generative AI Tool Used (Please List Each Separately)	Purpose of Use	Briefly Explain the Extent of Use
Chat GPT	Generate Template, edit and correct the code	In some cases, I would like to have a better view from a user's perspective.
GitHub	Publication of HTML	To enable me as the designer to share the Webpage via a link. So, I made GitHub my Host.

Student Name: Macario Cus

Student Id: 2015112273

Course Code: CMPS1134

Department: AINT

Signature: 

Date: 11/Nov/25