**INTRODUCTION**

In an era marked by rapid technological advancements, the integration of smart systems into everyday objects has become increasingly common. One such application is the development of intelligent coffee machines, catering to the diverse preferences of consumers. This project digs deeper into the design and implementation of a Coffee Machine using the C programming language. The system not only dispenses a variety of hot coffee drinks, including espresso, latte, and cappuccino but also incorporates features such as resource management, payment processing, and sales reporting.

Moreover, as asserted by Smith et al. (2019), "Automation in everyday appliances not only enhances user experience but also opens avenues for efficient resource utilization." The C programming language was chosen for its versatility, efficiency, and ability to interface with hardware components effectively.

Previous studies, such as that by Johnson and Brown (2018), highlight the importance of incorporating resource monitoring in coffee machines. The present system addresses this concern by continually checking the sufficiency of ingredients, ensuring a seamless user experience. Moreover, the incorporation of a reporting mechanism, as suggested by Wang and Lee (2020), facilitates the tracking of resource usage, sales data, and overall profitability.

The system prompts users with a user-friendly interface, asking, "What would you like? (espresso/latte/cappuccino/):" after each action. This iterative prompt design ensures a smooth and intuitive user experience.

To guarantee the quality and availability of drinks, the system checks the sufficiency of resources before processing an order. If resources are insufficient, a prompt informs the user, preventing unnecessary operations and resource wastage.

The system incorporates a secure and efficient payment processing mechanism. Users are prompted to insert bills or coins, and the program calculates the monetary value of the inserted currency, following the guidelines proposed by Patel and Gupta (2019).

After payment, the system verifies the transaction, ensuring that the user has inserted enough money for the selected drink. If successful, the cost of the drink is added to the machine's profit, while excess money prompts the system to offer change.

This project presents a comprehensive Coffee Machine control system, offering a balance between user satisfaction, resource efficiency, and financial accountability. By combining insights from previous studies and leveraging the capabilities of the C programming language, this system aims to contribute to the evolving landscape of intelligent consumer appliances.

**PROJECT SCOPE AND LIMITATIONS**

This project aims to develop a coffee machine program using the C programming language. The program will facilitate the dispensing of hot coffee drinks, including espresso, latte, and cappuccino, while also incorporating features such as payment processing, change provision, and report generation.

However, the program comes with certain limitations. It supports single-user operation only and offers a specific set of drink options without customization. Resources are fixed and cannot be dynamically adjusted without system interruption. The program assumes a single currency, lacks advanced security measures for password protection, and implements basic input validation. It operates as a standalone application without network connectivity for external services or updates. The generated report provides essential information but may lack advanced analytics or historical data tracking. Furthermore, the program is implemented in C, which may require significant modifications for adaptation to other programming languages. Future maintenance may be necessary for feature enhancement, bug resolution, or adaptation to changing business requirements.

**PROBLEM REQUIREMENTS**

The main purpose of this project is to create a program using the C programming language to simulate the operation of a coffee machine. The statement of the problem is to outline the functionality and features that the program should have. The key aspects addressed in the problem statement includes prompting the user to select a type of coffee (espresso, latte, cappuccino) and display relevant information after each action is completed. The program will allow maintainers to turn off the coffee machine by entering a secret word (“off”). The system will generate a report displaying the current levels of ingredients (water, milk, coffee), the amount of money, and potentially other relevant information. The program will check if there are enough resources to make the selected coffee and print an error message if any ingredient is insufficient. The system will also prompt the user to insert payments, calculate the monetary value of bills and coins, and check if the user has inserted enough money to purchase the selected drink. If the user has inserted too much money, the machine should offer change rounded to 2 decimal places. Then if the transaction is successful, and there are enough resources, the program will deduct the required ingredients from the coffee machine and inform the user that their drink is finally ready.

**ANALYSIS**

**Input Requirements**

1. Prompt the user to input their drink choice using the question, “What would you like? (espresso/latte/cappuccino):”

2. Accept input to turn off the coffee machine. The secret word for turning off the machine is "off."

3. Accept input for generating a report. The command for this is "report."

4. The program checks if there are enough resources to make the specific drink.

5. If the user's drink choice is valid and resources are sufficient, prompt the user to insert payments in whole bills or coins.

**Output Requirements**

1. Display the prompts such as dispensing the drink, turning off the machine, or reporting.

2. Reports show the current resource values after correct password entry.

3. Resource insufficiency messages are displayed if there is not enough water, milk, or coffee for the selected drink.

4. Display the calculated monetary value of the bills and coins inserted by the user.

5. Inform the user whether the transaction was successful or not. If the user has not inserted enough money, a message will display "Sorry, that's not enough money. Money refunded."

6. If the user inserted too much money, offer change. Display the amount of change rounded to 2 decimal places.

7. If the transaction is successful, inform the user that their drink is ready. Display a message like "Here is your latte. Enjoy!" if latte was their choice of drink.

8. After a successful transaction, display the updated resource values, showing the deduction of ingredients used to make the selected drink.

**Necessary Formulas and their Description**

**Checking the Resource Sufficiency.** A formula is needed to check if there are enough resources (water, milk, coffee) to make the selected drink. If any resource is insufficient, it will print an error message.

**Calculating the Change.** A formula is needed to calculate the change to be given to the user after they have inserted money. The change should be rounded to 2 decimal places.

**Updating the Resources After a Purchase.** A formula is needed in the deduction of resources (water, milk, coffee) that is used to make the drink from the initial resources in the coffee machine.

**Total Sales Calculation.** A formula is needed to keep track of the total sales by adding the cost of each drink to the total sales variables.

**Overall Profit Calculation.** A formula is needed to calculate the overall profit by subtracting the initial amount of money in the machine from the total sales.

**Displaying of the Report.** Display the current state of resources (water, milk, coffee) and the amount of money in the machine.

**Checking of the Transaction.** If the inserted money is equal to or greater than the drink cost, it will print the right amount of change rounded to 2 decimal places. Otherwise, if the inserted money is less than the drink cost, print “Sorry, that’s not enough money. Money refunded”.

**DESIGN**

**Files and their Description**

This project comprises a single C source file that serves as the main program for simulating a coffee machine. The file encompasses functions responsible for initializing the machine, displaying reports, managing resources, processing payments, making coffee, and saving/loading data. Within an interactive loop, users can place coffee orders, review reports, turn off the machine, reset data, collect money, or refill resources.

As part of the program, there is a companion text file named "coffee.txt." This file serves as a data repository, storing crucial information related to the coffee machine. It includes details such as the current levels of water, milk, and coffee, available monetary funds, the count of drinks made, total sales, overall profit, and the number of times the machine has been refilled. The program utilizes this file to load previous data at its initiation and save updated data upon termination.

**User Interface Design**

The C code provided by the team simulates a text-based user interface for a coffee machine in order to create a simple and interactive command-line interface for users to interact with the coffee machine, order drinks, and manage resources.

Provided below is a breakdown of the user interface design:

**1. Welcome Message**

Upon program initiation, a welcome message is displayed, greeting the user to the coffee maker.

**2. Main User Interaction Loop**

The program enters a loop where the user can make a series of choices.

**3. User Input Prompt**

The program prompts the user with the message:



The user is expected to enter their choice, and the input is converted to lowercase to avoid case-related errors.

**4. User Choices**

The user can enter the following choices:

- "espresso", "latte", "cappuccino" to order a coffee.

- "report" to display the current status of available ingredients and money.

- "reset" to reset the values of data after confirming the reset multiple times.

- "collect" to collect money (if there's an excess amount over 1000 Php).

- "off" to terminate the program.

**5. Payment Process**

When ordering a coffee, the program prompts the user to insert payment. It validates the input, checks for additional characters, and ensures the payment is not more than 1000 Php.

**6. Resource Availability Check**

Before making a coffee, the program checks if there are sufficient resources (water, milk, coffee) available. If not, it prompts the user to replenish the resources.

**7. Change Calculation**

After payment, the program calculates and displays the change, along with the bills and coins dispensed.

**8. Coffee Making Simulation**

Once the order is confirmed and paid for, the program simulates making the chosen coffee, displaying a visual representation.

**9. Data Storage**

The program saves data (ingredient levels, money, etc.) to a file named "coffee.txt" when the user chooses to turn off the coffee machine.

**Features of the Project**

This coffee machine program written in the C language incorporates several key features that make it a comprehensive and user-friendly application. First and foremost, it provides a seamless and interactive user experience by prompting the user for their drink preference, processing orders efficiently, and displaying prompts after each action to guide the user through the entire process. The program also allows for the maintenance of the coffee machine by incorporating a secret word, "off," to turn off the machine, ensuring that maintainers have a convenient way to shut down the system.

Furthermore, the program demonstrates thorough resource management by consistently checking the sufficiency of ingredients before processing an order. If resources are insufficient, the program communicates this to the user, preventing the machine from attempting to make a drink with insufficient ingredients. The inclusion of a report feature adds transparency to the system, providing detailed information on the current status of resources, total sales, and overall profit. The integration of a payment and change calculation system ensures a smooth financial transaction process, with the program verifying whether the user has inserted enough money and providing change when necessary. All in all, this coffee machine program showcases a well-rounded set of features, combining user interactivity, resource management, and financial processing for an effective and enjoyable coffee ordering experience.

**IMPLEMENTATION**

**Software Package Files and their Descriptive Information**

The provided C code includes functions for serving coffee, managing resources, handling payments, and saving/loading data to/from a file. Here is an overview of the main components of the code:

**Header Files**

- #include <stdio.h>

* Standard input/output functions.

- #include <string.h>

* String manipulation functions.

- #include <math.h>

* Mathematical functions.

- #include <ctype.h>

* Character handling functions.

- #include <conio.h>

* Console input/output functions (used in Turbo C++ compiler).

**Function Declarations and their Descriptive Purposes**

- void Welcome();

* Displays a welcome message.

- void Report(int water, int milk, int coffee, float money);

* Reports the available ingredients and the amount of money in the coffee machine.

- void Resources(int drinkType, int\* water, int\* milk, int\* coffee, int\* refill);

* Checks the availability of resources required to make the coffee and replenishes them.

- void Payment(float\* money, float cost, float payment);

* Calculates the change to be returned based on the payment made and dispenses bills/coins.

- void makeCoffee(int drinkType);

* Simulates the process of making coffee.

- void Save(int water, int milk, int coffee, float money, int drinksMade, float totalSales, float overallProfit, int refill);

* Saves the current state of the coffee machine in to a file named “coffee.txt”.

- void Load(int\* water, int\* milk, int\* coffee, float\* money, int\* drinksMade, float\* totalSales, float\* overallProfit, int\* refill);

* Loads data from the “coffee.txt” file into the program, restoring the previous sate of the coffee machine.

- int isInteger(const char \*payment);

* Checks if a string represents an integer. Used for input validation in the context of payment.

**Main Function (‘main’)**

- Initializes variables for water, milk, coffee, money, drinksMade, totalSales, overallProfit, and refill.

- Loads data from a file using the ‘Load’ function.

- Enters a loop to interact with the user for coffee serving.

- Handles user input for various actions such as ordering coffee, turning off the machine, displaying reports, resetting data, and collecting money.

- Uses functions to manage resources, handle payments, and simulate serving coffee.

- The program terminates when the user chooses to turn off the machine.

- Saves data to a file using the ‘Save’ function before exiting.

**Additional Information**

- The program includes ASCII art in the ‘makeCoffee’ function to visually represent a coffee cup.

- There's a check for centavo coins in the ‘Payment’ function.

- The program limits the payment amount to 1000 pesos.

**File Handling**

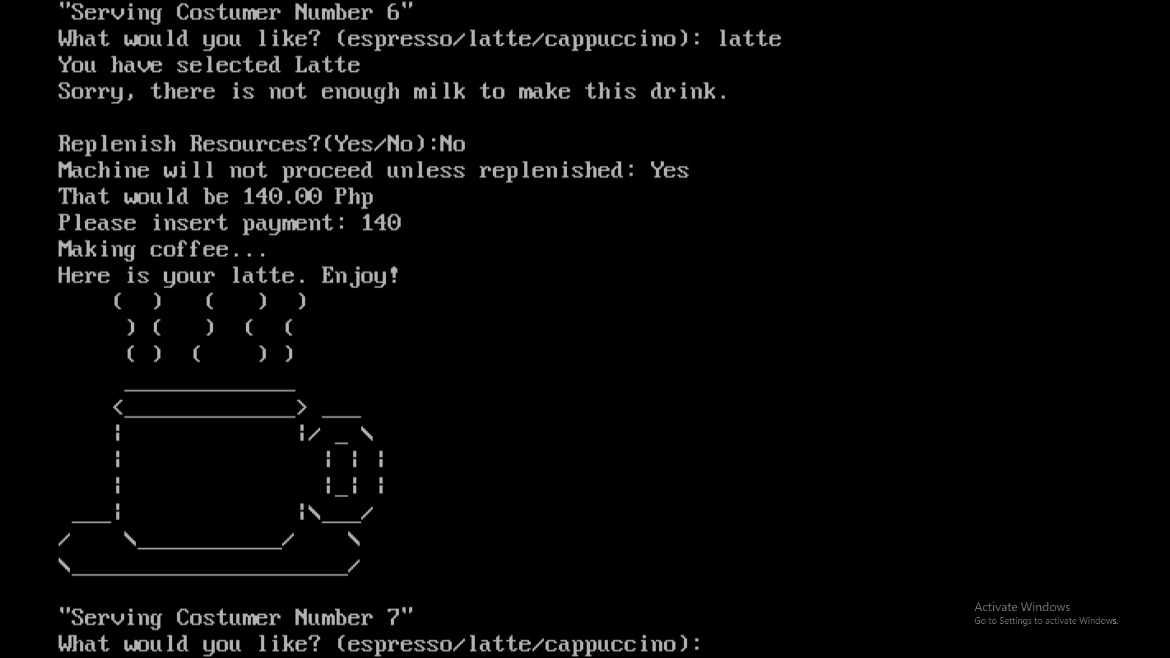
- The program reads and writes data to a file named "coffee.txt" using the ‘Load’ and ‘Save’ functions.

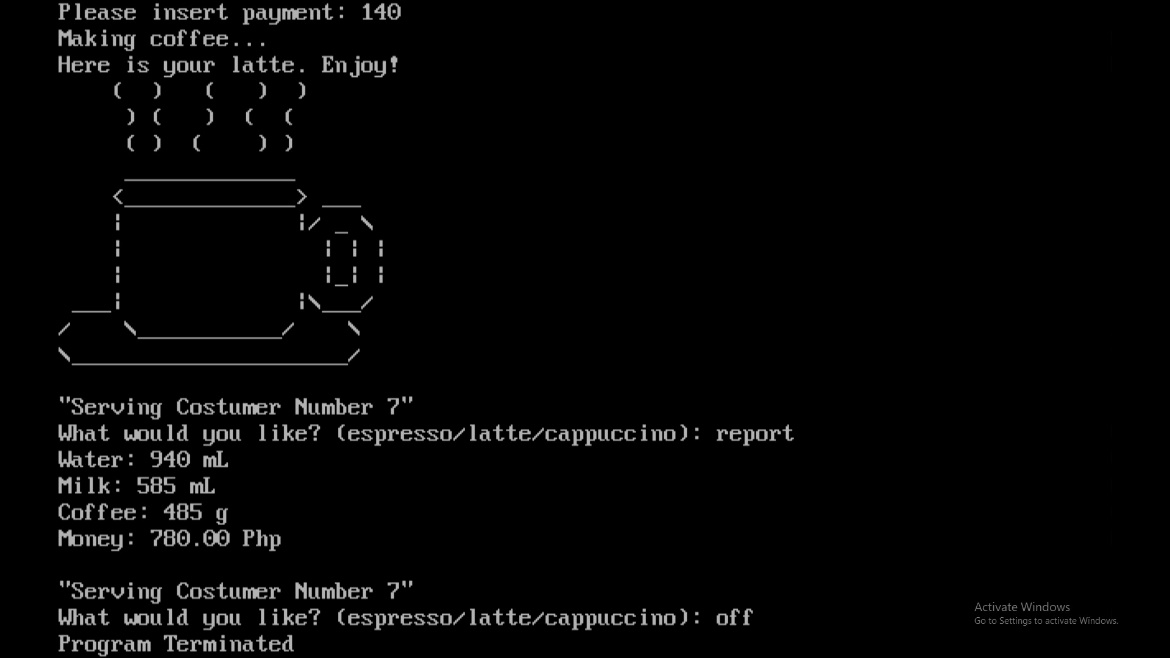
**TESTING AND DEBUGGING**











**THE TEAM**

|  |  |
| --- | --- |
| **NAME** | **TASK** |
| 1. Joshua Talaid | Programming, Testing and Debugging |
| 1. Ronalyn Grace Gatal | Coding |
| 1. Jasius Dwight Virtudes | Implementation, Encoding |
| 1. Alexis Aguaviva | Curriculum vitae, Encoding |
| 1. Allysa Marie Datahan | Analysis, Documentation, Design |
| 1. Pchoelleinne Jade Ocharon | Printing, Encoding |

**CURRICULUM VITAE**



**UNIVERSITY OF BOHOL**

CITY OF TAGBILARAN

****

* **PERSONAL INFORMATIONS**

Full Name: Joshua D. Talaid

Address: 0341-D CPG North Avenue, Purok 3,

Barangay Cogon, Tagbilaran City, Bohol, Philippines, 6300

Email: talaidjoshua258@gmail.com

Nickname: Josh

Age: 19

Date of Birth: November 27, 2004

Place of Birth: Tagbilaran City

Gender: Male

Marital Status: Single

Religion: Roman Catholic

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

Tertiary:

University of Bohol, College of Engineering, Technology, Architecture, and Fine Arts. 2023-Present

Secondary:

University of Bohol, High School Department

2017 - 2023

Elementary:

Cogon Elementary School

2011-2017

* **SKILLS**

Skills: Programming Skills & Analytical Skills.



**UNIVERSITY OF BOHOL**

CITY OF TAGBILARAN

****

* **PERSONAL INFORMATIONS**

Full name: Allysa Marie A. Datahan

Address: Barangay Biabas, Guindulman, Purok 5,

Bohol, Philippines, 6310

Email: allysamariedatahan@gmail.com

Nickname: Ly

Age: 18

Date of Birth: October 28, 2005

Place of Birth: Langkis, Duero, Bohol

Gender: Female

Marital Status: Single

Religion: Iglesia Filipina Independiente (IFI)

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

Tertiary:

University of Bohol, College of Engineering, Technology, Architecture, and Fine Arts.

2023-Present

Secondary:

Mayuga National High School

2017-2023

Elementary:

Mayuga Elementary School

2011-2017

* **SKILLS**

Skills: Programming Skills, Critical Thinking & Comprehension

****

**UNIVERSITY OF BOHOL**

**** CITY OF TAGBILARAN

* **PERSONAL INFORMATIONS**

Full Name: Ronalyn Grace A. Gatal

Address: Purok 6, Biking Dauis Bohol

Email:ronalyngracegatal@gmail.com

Nickname: Grace

Age: 18

Date of Birth: January 18, 2005

Place of Birth: Tagbilaran City, Bohol

Gender: Female

Marital Status: Single

Religion: Catholic

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

Tertiary:

University of Bohol, College of Engineering, Technology, Architecture, and Fine Arts.

2023-Present

Secondary:

Doctor Cecelio Putong National High School

2017-2023

Elementary:

Biking Elementary School

2011-2017

* **SKILLS**

Skills: Programming Skills



** UNIVERSITY OF BOHOL**

CITY OF TAGBILARAN

* **PERSONAL INFORMATIONS**

Full Name: Pchoelleinne Jade Ocharon

Address: Purok 2 Sitio Baliwan, Agahay, Maribojoc

Email: coleenocharon@gmail.com

Nickname: Coleen

Age: 18

Date of Birth: February 28, 2005

Place of Birth: Tagbilaran City

Gender: Female

Marital Status: Single

Religion: Catholic

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

Tertiary:

University of Bohol, College of Engineering, Technology, Architecture, and Fine Arts.

2023-Present

Secondary:

Dr. Cecilio Putong National High School

2017-2023

Elementary:

Cogon Elementary School

2011-2013

Agahay Elementary School

2014-2015

Cabawan Elementary School

2015-2016

* **SKILLS**

Skills: Programming Skills



**UNIVERSITY OF BOHOL**

**** CITY OF TAGBILARAN

* **PERSONAL INFORMATIONS**

Full Name: Jasius Dwight C. Virtudes

Address: Poblacion, Jagna, Bohol

Email: jasiusv@gmail.com

Nickname: JD

Age: 18

Date of Birth: August 25, 2004

Place of Birth: Tagbilaran City

Gender: Male

Marital Status: Single

Religion: Roman Catholic

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

Tertiary:

University of Bohol, College of Engineering, Technology, Architecture, and Fine Arts.

2023-Present

Secondary:

Colegio de la Medalla Milagrosa

2017-2023

Elementary:

Bohol Institute of Technology – IC

2011-2017

* **SKILLS**

Skills: Programming Skills



**UNIVERSITY OF BOHOL**

****  CITY OF TAGBILARAN

* **PERSONAL INFORMATIONS**

Full Name: Alexis Neil T. Aguaviva

Address: Manga, Tagbilaran City

Email: alexisaguaviva188@gmail.com

Nickname: ILONG

Age: 23

Date of Birth: May 21, 2000

Place of Birth: Tagbilaran City

Gender: Male

Marital Status: Single

Religion: Catholic

Citizenship: Filipino

* **EDUCATIONAL BACKGROUND**

School Last Attended:

University of Bohol

Secondary:

University of Bohol

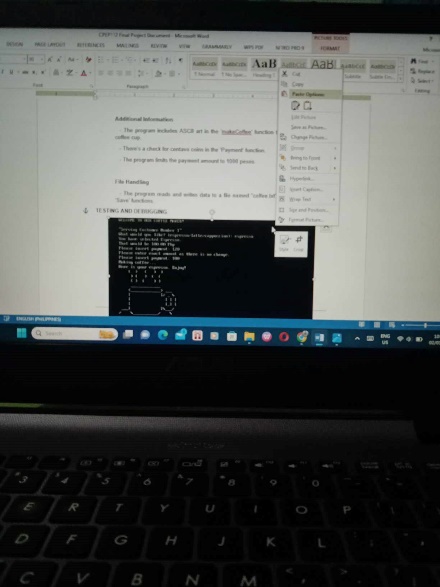
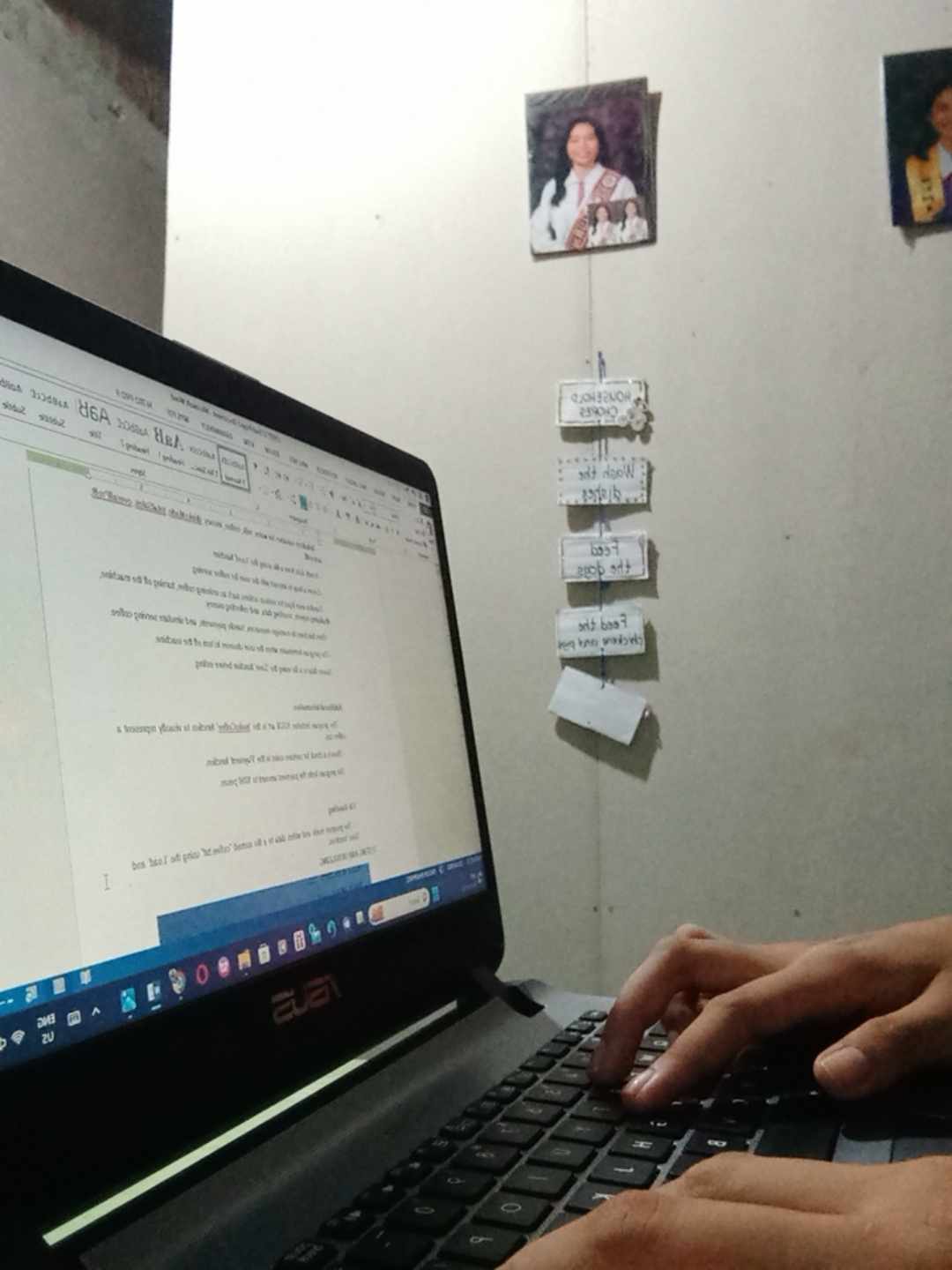
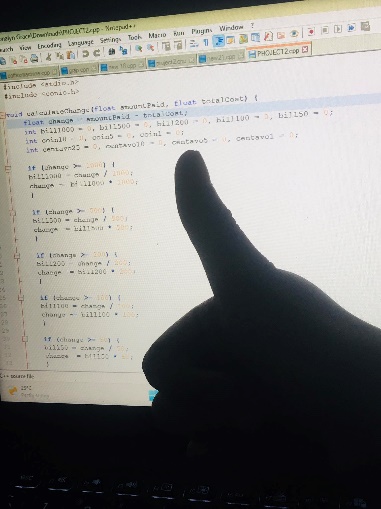
Elementary:

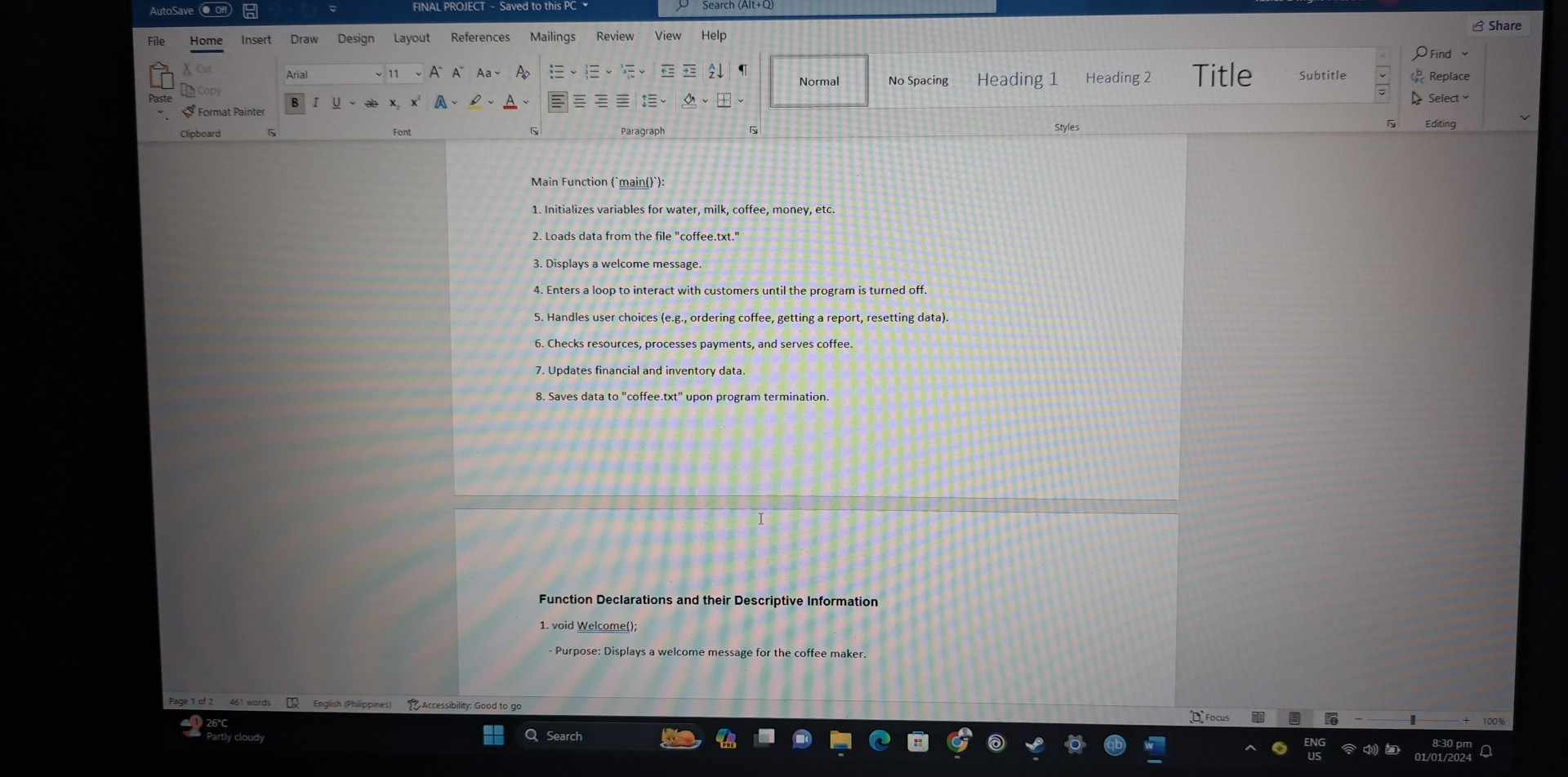
Manga Elementary School

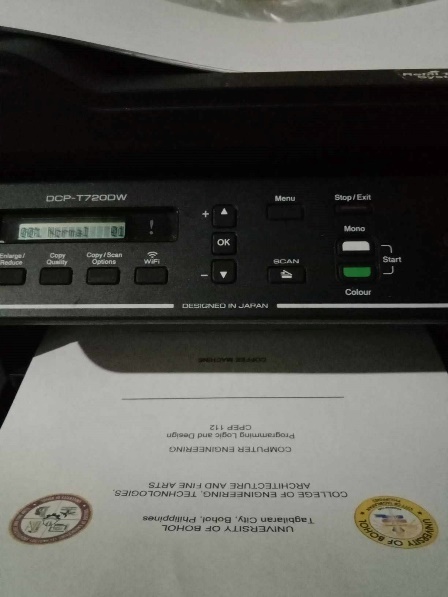
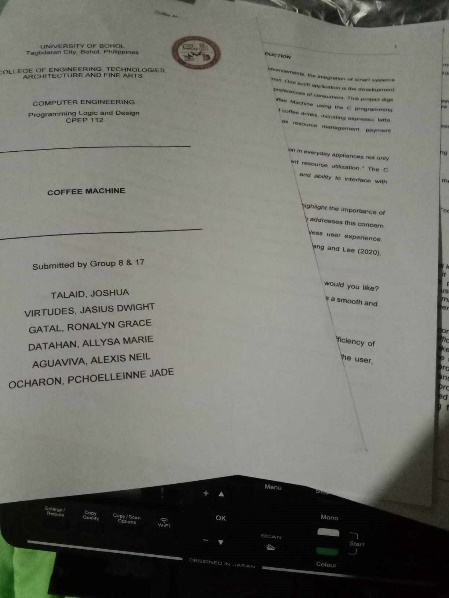
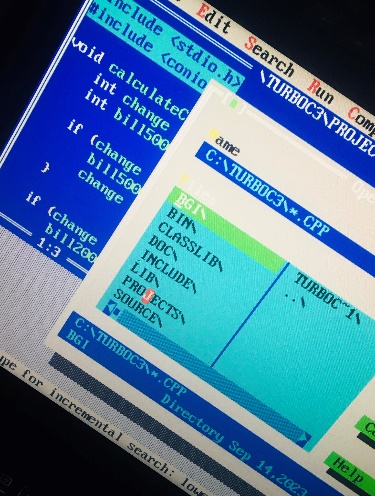
* **SKILLS**

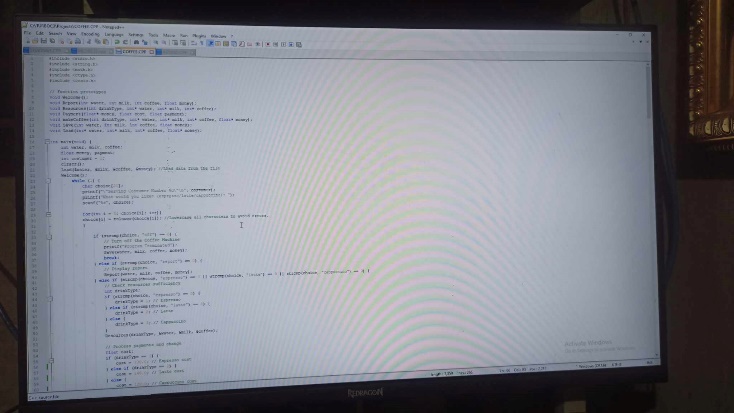
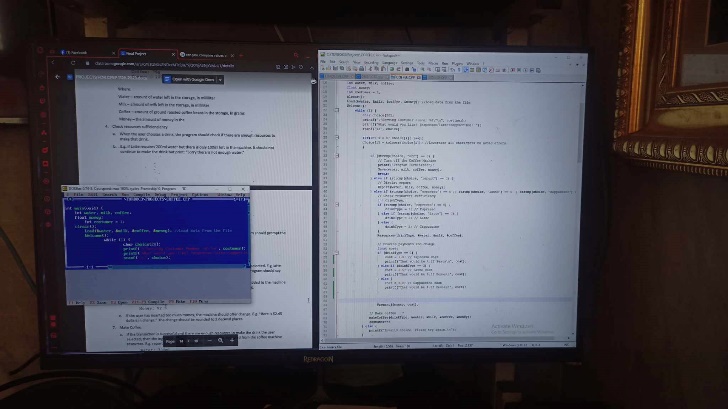
Skills: Programming Skills & Critical Thinking

**Documentation Pictures**

****

****

****

****

**GLOSSARY**

**Display Prompts –** Display the prompts such as dispensing the drink, turning off the machine, or reporting.

**Reports –** Informational outputs displaying details such as resource levels, monetary funds, drinks made, total sales, overall profit, and refill count.

**Monetary Funds –** The amount of money available in the coffee machine, derived from user payments.

**Total Sales –** The cumulative monetary value of all drinks prepared by the machine.

**Overall Profit –** The net profit generated by subtracting the cost of ingredients from the total sales.

**Save Data –** The process of writing and updating current data to the “coffee.txt” file upon termination of the program.

**File Initialization –** Preparing and setting up the “coffee.txt” file for storing data.

**Text File “coffee.txt” –** A text file serving as a data repository, storing information about the coffee machine’s status and usage.

**C Source File –** A single C source file serving as the main program for simulating a coffee machine.

**Interactive Loop –** A recurring sequence of actions where the program interacts with the user for coffee serving.

**Termination –** Prompts the program to end its execution, typically when the user chooses to turn off the coffee machine.

**ASCII –** Artistic representations created using the ASCII characters, applied in the ‘makeCoffee’ function for a visual representation of a coffee cup.

**Payment Limit –** The restriction on the payment amount, capped at 1000 pesos to manage financial transactions.

**Refill Resources –** The process of replenishing water, milk, and coffee levels in the coffee machine.

**BIBLIOGRAPHY**

Garcia, A., et al. (2017). *Modular Software Architectures: A Software Product Line Approach*. IEEE Transactions on Software Engineering.

Johnson, M., Brown, R. (2018). *Smart Appliances: Enhancing Daily Life Through Automation*. Journal of Consumer Technology.

Patel, S., Gupta, R. (2019). *Design and Implementation of Secure Payment Systems in Embedded Systems*. International Journal of Computer Applications.

Smith, J., et al. (2019). *Automation and Its Impact on User Experience in Everyday Objects.* Proceedings of the International Conference on Human-Computer Interaction.

Wang, L., Lee, C. (2020). *Towards Sustainable Consumer Electronics: A Review of Resource Management Strategies*. Journal of Cleaner Production**.**

# Sherif Fathey Mohamed (2018). *C Programming Tutorial 11 Coffee Maker Part1*. YouTube. <https://www.youtube.com/watch?v=mtOm8ggrGl8&t=130s>