Program: Budget Tracker

Course:

CPE007 – Programming Logic and Design

Section:

CPE11S6

Instructor:

Engr. Robin Valenzuela

Group 1

Members:

Simon Patrick P. Lapuz

Bien Macaraeg

Thea Padilla

Gian Carlo Veruen

Date:

November 13, 2025

Objectives

- To allow the user to input their monthly income, expenses, and spending limit.
- Calculate how much the user will have saved based on their monthly income and spending limit.
- To allow the user to calculate reduced spending based on percentage for them to see how much they can save instead.

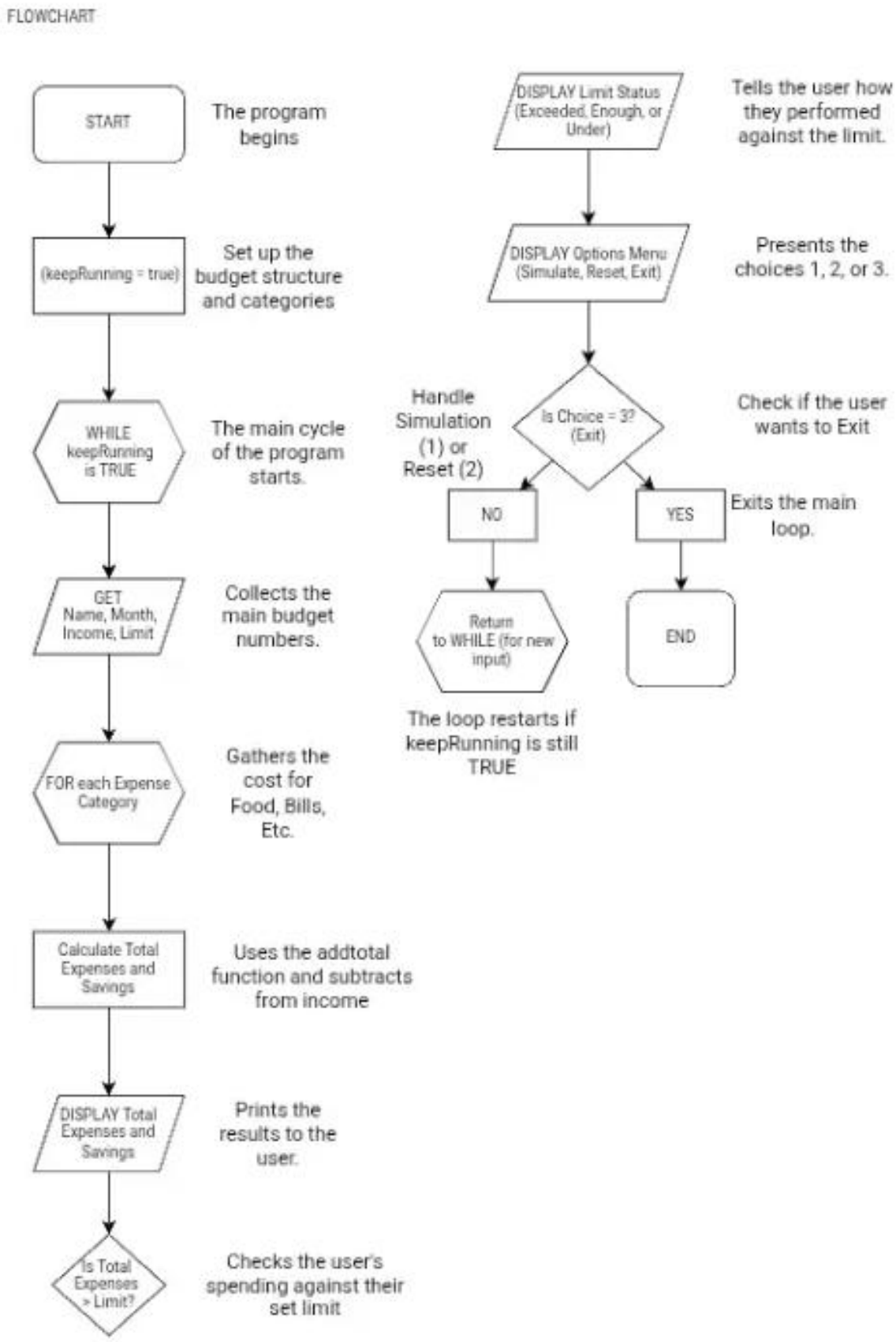
Statement of the Problem

As we all know, managing our finances may prove to be quite a challenge. Many people often have difficulties keeping track of their monthly savings and expenses which usually leads to poor management of their finances. In order to give these people a helping hand, our group will be developing a C++ program that allows users to efficiently track their monthly savings and expenses.

Pseudocode

```
START

WHILE program is running
  DISPLAY "MONTHLY BUDGET TRACKER"

  INPUT name
  INPUT month
  INPUT income
  INPUT spending limit

  FOR each of the 5 categories (food, transportation, bills, recreation, others)
    INPUT expense
  END FOR

  total_expenses = sum of all expenses
  savings = income - total_expenses

  DISPLAY total_expenses and savings

  IF total_expenses > limit THEN
    DISPLAY "You exceeded your spending limit by"
  ELSE IF total_expenses == limit THEN
    DISPLAY "Perfect budgeting! You've exactly met your spending limit"
  ELSE
    DISPLAY "Great job! You saved, You are under your limit!"
  END IF

  DISPLAY options:
    1 - Simulate reduced spending
    2 - Reset information
    3 - Exit program

  INPUT choice
  IF choice == 1 THEN
    INPUT reduction_percent
    new_total = total_expenses * (1 - reduction_percent / 100)
    new_savings = income - new_total
    DISPLAY new_total and new_savings
  ELSE IF choice == 2 THEN
    RESTART input
  ELSE IF choice == 3 THEN
    DISPLAY "Thank you for using the Budget Tracker. Keep it up and be financially smart"
    STOP program
  ELSE

      DISPLAY "Invalid choice. Please try again."

WHILE backToInput is false AND keepRunning is true
  repeat options menu
END WHILE

IF keepRunning is false THEN
  EXIT program
  END IF
END WHILE

END
```

Flowchart


FLOWCHART
START
The program begins
(keepRunning = true)
Set up the budget structure and categories
WHILE keepRunning is TRUE
The main cycle of the program starts.
GET Name, Month, Income, Limit
Collects the main budget numbers.
FOR each Expense Category
Gathers the cost for Food, Bills, Etc.
Calculate Total Expenses and Savings
Uses the addtotal function and subtracts from income
DISPLAY Total Expenses and Savings
Prints the results to the user.
Is Total Expenses > Limit?
Checks the user's spending against their set limit
DISPLAY Limit Status (Exceeded, Enough, or Under)
Tells the user how they performed against the limit.
DISPLAY Options Menu (Simulate, Reset, Exit)
Presents the choices 1, 2, or 3.
Handle Simulation (1) or Reset (2)
Is Choice = 3? (Exit)
Check if the user wants to Exit
NO
YES
Exits the main loop.
Return to WHILE (for new input)
END
The loop restarts if keepRunning is still TRUE

Program

```cpp
#include <iostream>
#include <string>
using namespace std;

// function to calculate total expenses for the month
double addtotal (double a, double b, double c, double d, double e) {
    return (a + b + c + d + e);
}

int main () {
    // declare struct to include information and expenses
    struct budget {
        string name;
        string month;
        double income;
        double limit;
        double expenses [5];
    };

    budget user;
    string categories [5] = {"food", "transportation", "bills", "recreation", "others"}; // declare string array for the different categories of
    double totalexp; // declare variable that displays the total amount of expenses

    bool keepRunning = true;
    while (keepRunning) {

        // ====== INPUT SECTION ======
        cout << "==MONTHLY BUDGET TRACKER AND CALCULATOR==" << endl;

        // Use >> ws to consume any leftover whitespace so getline works reliably
        cout << "Enter your name: ";
        getline(cin >> ws, user.name);

        cout << "Enter the month for your budget: ";
        cin >> user.month;
        cout << "Enter your monthly income: ";
        cin >> user.income;
        cout << "Enter your monthly spending limit: ";
        cin >> user.limit;

        cout << "Enter your monthly expenses for the following categories: " << endl;
        for (int i = 0; i < 5; i++) { // for loop to input the expenses for each category
            cout << categories[i] << " : ";
            cin >> user.expenses[i];
        }


        // ====== OUTPUT / CALCULATION ======
        cout << "\n==BUDGET CALCULATION REPORT===" << endl;
        totalexp = addtotal(user.expenses[0], user.expenses[1], user.expenses[2], user.expenses[3], user.expenses[4]);
        cout << "Total Expenses for the month of " << user.month << ": " << totalexp << endl;

        double savings = user.income - totalexp;
        cout << "Your total savings: " << savings << endl;

        // Check spending against limit
        if (totalexp > user.limit) {
            cout << "You have exceeded your spending limit by " << totalexp - user.limit << endl;
        }
        else if (totalexp == user.limit) {
            cout << "Perfect budgeting! You've exactly met your spending limit." << endl;
        }
        else {
            cout << "Great job! You saved " << user.limit - totalexp << " under your spending limit." << endl;
        }


        // ====== OPTIONS MENU ======
        int choice;
        bool backToInput = false;
        do {
            cout << "\n=== OPTIONS MENU ===" << endl;
            cout << "1. Simulate reduced spendings" << endl;
            cout << "2. Reset information" << endl;
            cout << "3. Exit app" << endl;
            cout << "Enter your choice (1-3): ";
            cin >> choice;

            if (choice == 1) {
                double reducePercent;
                cout << "\nEnter percentage to reduce each spending (e.g., enter 10 for 10%): ";
                cin >> reducePercent;

                double newTotal = 0;
                for (int i = 0; i < 5; i++) {
                    newTotal += user.expenses[i] * (1 - (reducePercent / 100.0));
                }
```

```
            double newSavings = user.income - newTotal;
            cout << "\nIf you reduce your expenses by " << reducePercent << "%, your new total expenses will be: " << newTotal << endl;
            cout << "Your new savings will be: " << newSavings << endl;
        }
        else if (choice == 2) {
            cout << "\nResetting information...\n" << endl;
            backToInput = true;
        }
        else if (choice == 3) {
            cout << "\nThank you for using the Budget Tracker. Keep it up and be financially smart, " << user.name << "!" << endl;
            keepRunning = false;
            break;
        }
        else {
            cout << "\nInvalid choice. Please try again." << endl;
        }
    }
    while (!backToInput && keepRunning);

    // loop will either restart for new input (reset), or exit if keepRunning==false
    if (!keepRunning) break;
    cout << "-------------------------------------\n";
} // end while keepRunning

return 0;
}
```

Outputs

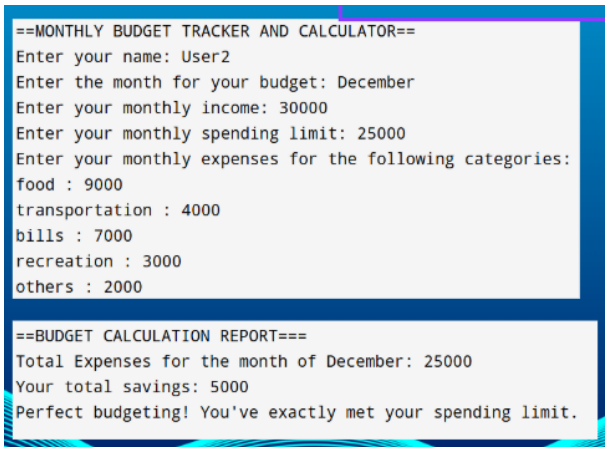
Output (Spend under the spending limit)

```
==MONTHLY BUDGET TRACKER AND CALCULATOR==
Enter your name: User1
Enter the month for your budget: November
Enter your monthly income: 30000
Enter your monthly spending limit: 25000
Enter your monthly expenses for the following categories:
food : 8000
transportation : 3000
bills : 6000
recreation : 2000
others : 1500

==BUDGET CALCULATION REPORT===
Total Expenses for the month of November: 20500
Your total savings: 9500
Great job! You saved 4500 under your spending limit.
```

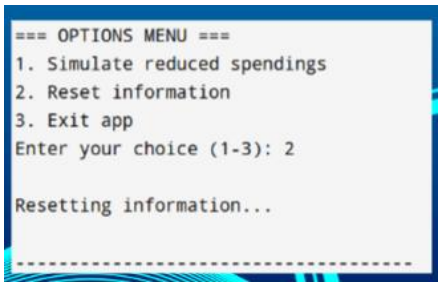
Output (Reset information)

```
=== OPTIONS MENU ===
1. Simulate reduced spendings
2. Reset information
3. Exit app
Enter your choice (1-3): 2

Resetting information...

-------------------------------------
```

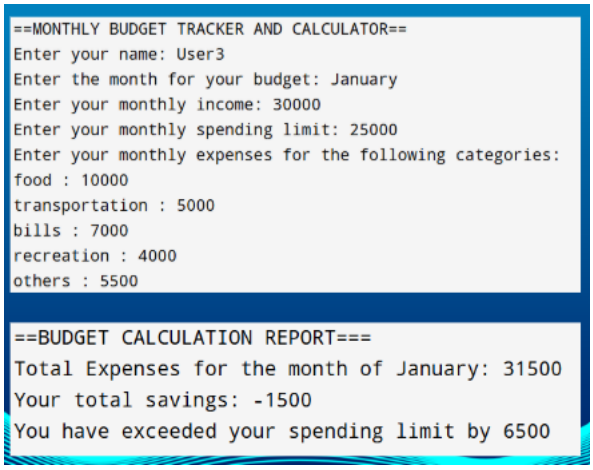
Output (Exactly meet the spending limit)



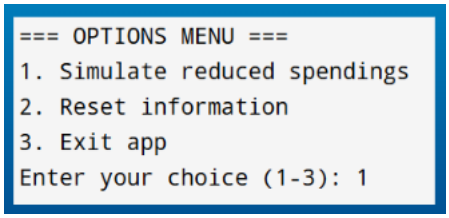
Output (Reset information)

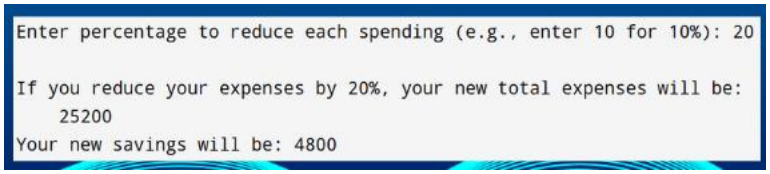


Output (Exceeded the spending limit)

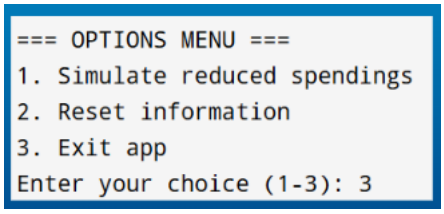


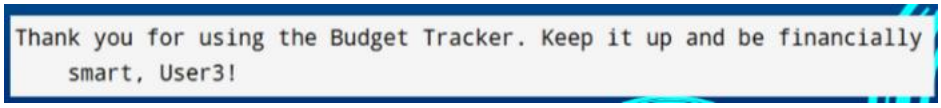
Output (Simulate reduced spendings)





Output (Exit and thank you message)





Output (Invalid choice)

```
=== OPTIONS MENU ===
1. Simulate reduced spendings
2. Reset information
3. Exit app
Enter your choice (1-3): 4

Invalid choice. Please try again.
```

Conclusion

We created a C++ program that lets users input their monthly income, expenses, and spending limit and our program will calculate their total savings based on their expenses. Additionally, we also give them the option to simulate how much they could save if they were to reduce their expenses to a specified percentage. This will help people keep track of where their money is spent for the month and it can also help people better manage their expenses which creates an opportunity for them to efficiently manage their expenses and grow their savings.