Electricity/Gas Bill Calculator

Overview

This C++ console application calculates electricity bills split between two floors (First Floor and Ground Floor) based on meter readings. It supports two modes:

- **Late Reading:** Calculates average daily usage when meter readings are late and estimates consumption.
- **Normal Reading:** Calculates consumption based on regular meter readings.

It then splits the total bill amount proportionally between the two floors and adds taxes if applicable. Finally, it outputs a detailed report saved to a text file (`bill_report.txt`) and opens it automatically.

Features

- Input validation for meter readings.
- Percentage and cash split between floors.
- Handles taxes and divides them evenly.
- Saves a detailed bill report in a text file.
- Colored console output for better UX (Windows only).
- Supports late and normal reading modes.

Requirements

- Windows OS (for colored console and `ShellExecute` to open the file).
- C++ compiler supporting C++11 or later.
- `<windows.h>` and `<fstream>` libraries.

Code

```
#include <iostream>
#include <windows.h>
#include <fstream>
using namespace std;

int main() {
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    SetConsoleTextAttribute(hConsole, 10);  // Green text

    long int Bill, Wmeter, First_Floor, Ground_Floor;
    int ToPay, P1, P2, Tax;
    int Reading;
    long int LateRead;
    int LateDays;
    long int AvgLatePerDay;
    int NumOfLateDays;
```

```
    long int LateWmeter;
    long int LastWhiteLate;
    long int FinalLateReading;
    long int CurBill;
    long int BlueFinal;

    ofstream out("bill_report.txt");

    cout << "\n\t=====================================\n";
    cout << "\t     Electricity Bill Calculator\n";
    cout << "\t=====================================\n";

    cout << "\n\tChoose One Option\n\t 1 for Late Reading \n\t 2 for Normal
Reading" << endl;
    cin >> Reading;

    if (Reading == 1) {
        cout << "\n\tEnter the Current reading (White): " << endl;
        cin >> LateRead;
        if(LateRead < 0) {
            cout << "Error: Current white reading can’t be negative!\n";
            return 1;
        }

        cout << "\n\tEnter the Last reading(White):  " << endl;
        cin >> LastWhiteLate;
        if(LastWhiteLate < 0) {
            cout << "Error: Last white reading can’t be negative!\n";
            return 1;
        }

        if(LateRead < LastWhiteLate) {
            cout << "Error: Current white reading can’t be less than last white
reading!\n";
            return 1;
        }

        FinalLateReading = LateRead - LastWhiteLate;

        cout << "\n\tEnter Days:" << endl;
        cin >> LateDays;
        if(LateDays <= 0) {
            cout << "Error: Days must be positive!\n";
            return 1;
        }

        AvgLatePerDay = FinalLateReading / LateDays;

        cout << "\n\tAverage use per day:" << AvgLatePerDay << endl;
        cout << "\n\t number of days to calculate for: " << endl;
        cin >> NumOfLateDays;
        if(NumOfLateDays <= 0) {
            cout << "Error: Number of late days must be positive!\n";
            return 1;
```

```
        }

        LateWmeter = AvgLatePerDay * NumOfLateDays;

        cout << "\n\t Enter the Last reading from Bill: ";
        cin >> Bill;
        if(Bill < 0) {
            cout << "Error: Last reading from bill can’t be negative!\n";
            return 1;
        }

        cout << "\n\t Enter the Current reading from Bill: ";
        cin >> CurBill;
        if(CurBill < 0) {
            cout << "Error: Current reading from bill can’t be negative!\n";
            return 1;
        }

        if(CurBill < Bill) {
            cout << "Error: Current reading can’t be less than last reading!\n";
            return 1;
        }

        cout << "\n\t New Reading is: " << endl;
        BlueFinal = CurBill - Bill;
        cout << BlueFinal;

        First_Floor = LateWmeter;
        Ground_Floor = BlueFinal - First_Floor;

        cout << "\n\t--------------------------------------\n";
        cout << "\tFirst Floor consumption is: " << First_Floor << " units\n";
        cout << "\tGround Floor consumption is: " << Ground_Floor << " units\n";

        cout << "\n\tEnter total Cash (Rs): ";
        cin >> ToPay;
        if(ToPay < 0) {
            cout << "Error: Total cash can’t be negative!\n";
            return 1;
        }

        P1 = (First_Floor * 100) / BlueFinal;
        P2 = (Ground_Floor * 100) / BlueFinal;

        float FirstCash = ((float)First_Floor / BlueFinal) * ToPay;
        float GroundCash = ((float)Ground_Floor / BlueFinal) * ToPay;

        cout << "\n\t--------------------------------------\n";
        cout << "\tPercent Split:\n";
        cout << "\tFirst Floor: " << P1 << "%\n";
        cout << "\tGround Floor: " << P2 << "%\n";

        cout << "\n\tBill Split:\n";
        cout << "\tFirst Floor has to pay: Rs." << FirstCash << endl;
```

```
        cout << "\tGround Floor has to pay: Rs." << GroundCash << endl;

        cout << "\n\tPlease enter Taxes (if any): Rs. ";
        cin >> Tax;
        if(Tax < 0) {
            cout << "Error: Taxes can't be negative!\n";
            return 1;
        }

        int Taxes = Tax / 2;

        cout << "\n\tTax per floor: Rs." << Taxes << endl;

        cout << "\n\t========== Final Payment ==========\n";
        cout << "\tFirst Floor Total: Rs." << FirstCash + Taxes << endl;
        cout << "\tGround Floor Total: Rs." << GroundCash + Taxes << endl;
        cout << "\t====================================\n";

        SetConsoleTextAttribute(hConsole, 7);

        out << "===== GAS Bill Report =====\n";
        out << "Total Bill Units: " << BlueFinal << "\n";
        out << "White Meter Reading (First Floor): " << LateWmeter << "\n";
        out << "First Floor Units: " << First_Floor << "\n";
        out << "Ground Floor Units: " << Ground_Floor << "\n";
        out << "Total Cash: Rs. " << ToPay << "\n\n";

        out << "First Floor Percentage: " << P1 << "%\n";
        out << "Ground Floor Percentage: " << P2 << "%\n";
        out << "First Floor Bill: Rs. " << FirstCash << "\n";
        out << "Ground Floor Bill: Rs. " << GroundCash << "\n";
        out << "Tax per Floor: Rs. " << Taxes << "\n\n";
        out << "=== Final Payable ===\n";
        out << "First Floor: Rs. " << FirstCash + Taxes << "\n";
        out << "Ground Floor: Rs. " << GroundCash + Taxes << "\n";

        out.close();

        cout << "\n\t? Report saved to 'bill_report.txt'\n";

        ShellExecute(0, "open", "bill_report.txt", NULL, NULL, SW_SHOWNORMAL);

    } else {
        long int prevBill, currBill;
        long int prevWhite, currWhite;

        cout << "\n\t Enter previous reading from Bill: ";
        cin >> prevBill;
        if(prevBill < 0) {
            cout << "Error: Previous bill reading can't be negative!\n";
            return 1;
        }

        cout << "\n\t Enter current reading from Bill: ";
```

```
        cin >> currBill;
        if(currBill < 0) {
            cout << "Error: Current bill reading can’t be negative!\n";
            return 1;
        }

        if (currBill < prevBill) {
            cout << "\n\tError: Current reading cannot be less than previous
reading.\n";
            return 1;
        }

        cout << "\n\t Enter previous reading from (White) Meter: ";
        cin >> prevWhite;
        if(prevWhite < 0) {
            cout << "Error: Previous white meter reading can’t be negative!\n";
            return 1;
        }

        cout << "\n\t Enter current reading from (White) Meter: ";
        cin >> currWhite;
        if(currWhite < 0) {
            cout << "Error: Current white meter reading can’t be negative!\n";
            return 1;
        }

        if (currWhite < prevWhite) {
            cout << "\n\tError: Current reading cannot be less than previous
reading.\n";
            return 1;
        }

        long int billConsumption = currBill - prevBill;
        long int whiteConsumption = currWhite - prevWhite;

        First_Floor = whiteConsumption;
        Ground_Floor = billConsumption - whiteConsumption;

        cout << "\n\t--------------------------------------\n";
        cout << "\tFirst Floor consumption is: " << First_Floor << " units\n";
        cout << "\tGround Floor consumption is: " << Ground_Floor << " units\n";

        cout << "\n\tEnter total Cash (Rs): ";
        cin >> ToPay;
        if(ToPay < 0) {
            cout << "Error: Total cash can’t be negative!\n";
            return 1;
        }

        P1 = (First_Floor * 100) / billConsumption;
        P2 = (Ground_Floor * 100) / billConsumption;

        float FirstCash = ((float)First_Floor / billConsumption) * ToPay;
        float GroundCash = ((float)Ground_Floor / billConsumption) * ToPay;
```

```
        cout << "\n\t--------------------------------------\n";
        cout << "\tPercent Split:\n";
        cout << "\tFirst Floor: " << P1 << "%\n";
        cout << "\tGround Floor: " << P2 << "%\n";

        cout << "\n\tBill Split:\n";
        cout << "\tFirst Floor has to pay: Rs." << FirstCash << endl;
        cout << "\tGround Floor has to pay: Rs." << GroundCash << endl;

        cout << "\n\tPlease enter Taxes (if any): Rs. ";
        cin >> Tax;
        if(Tax < 0) {
            cout << "Error: Taxes can't be negative!\n";
            return 1;
        }

        int Taxes = Tax / 2;

        cout << "\n\tTax per floor: Rs." << Taxes << endl;

        cout << "\n\t========== Final Payment ==========\n";
        cout << "\tFirst Floor Total: Rs." << FirstCash + Taxes << endl;
        cout << "\tGround Floor Total: Rs." << GroundCash + Taxes << endl;
        cout << "\t======================================\n";

        out << "===== GAS Bill Report =====\n";
        out << "Total Bill Units: " << billConsumption << "\n";
        out << "White Meter Reading (First Floor): " << whiteConsumption << "\n";
        out << "First Floor Units: " << First_Floor << "\n";
        out << "Ground Floor Units: " << Ground_Floor << "\n";
        out << "Total Cash: Rs. " << ToPay << "\n\n";

        out << "First Floor Percentage: " << P1 << "%\n";
        out << "Ground Floor Percentage: " << P2 << "%\n";
        out << "First Floor Bill: Rs. " << FirstCash << "\n";
        out << "Ground Floor Bill: Rs. " << GroundCash << "\n";
        out << "Tax per Floor: Rs. " << Taxes << "\n\n";
        out << "=== Final Payable ===\n";
        out << "First Floor: Rs. " << FirstCash + Taxes << "\n";
        out << "Ground Floor: Rs. " << GroundCash + Taxes << "\n";

        cout << "\n\t? Report saved to 'bill_report.txt'\n";

        ShellExecute(0, "open", "bill_report.txt", NULL, NULL, SW_SHOWNORMAL);
    }

    return 0;
}
```

Output

```
        ======================================
             Electricity Bill Calculator
        ======================================

        Choose One Option
         1 for Late Reading
         2 for Normal Reading
2

         Enter previous reading from Bill: 220000

         Enter current reading from Bill: 420000

         Enter previous reading from (White) Meter: 120000

         Enter current reading from (White) Meter: 180000

        --------------------------------------
        First Floor consumption is: 60000 units
        Ground Floor consumption is: 140000 units

        Enter total Cash (Rs): 5000

        --------------------------------------
        Percent Split:
        First Floor: 30%
        Ground Floor: 70%

        Bill Split:
        First Floor has to pay: Rs.1500
        Ground Floor has to pay: Rs.3500

        Please enter Taxes (if any): Rs. 2000

        Tax per floor: Rs.1000

        ============ Final Payment ============
```

File

```
===== GAS Bill Report =====
Total Bill Units: 200000
White Meter Reading (First Floor): 60000
First Floor Units: 60000
Ground Floor Units: 140000
Total Cash: Rs. 5000

First Floor Percentage: 30%
Ground Floor Percentage: 70%
First Floor Bill: Rs. 1500
Ground Floor Bill: Rs. 3500
Tax per Floor: Rs. 1000

=== Final Payable ===
First Floor: Rs. 2500
Ground Floor: Rs. 4500

```