

ENGG*1410 (01011), 30th September, 2025

Lab 1 and 2: Implementing a pseudocode and Mastering VS Code IDE

Group 121

Pratham Lonial
&
Ishaandeep Basra

Table of Contents:

- **Lab 1**

- Part 1: Printing
- Part 2: Bill Splitting Program
- Part 3: Debugging

- **Lab 2**

- Part 1 Setting up vs Code:
- Part 2 Understanding vs Code:

Part 1: Printing

Problem Statement

The problem for part was to use the ‘printf’ command and use escape sequences to print out special characters and symbols.

Solution

To solve this problem, the instructions in the lab pdf were followed to print out the special characters and symbols.

Conclusion and self assessment

We learned the use of special characters and how to print and use each one of them.

Code

```
Lab 1 > C lab1print.c > main(void)
1  #include <stdio.h>
2
3  int main(void){
4      printf("C uses escape sequences for a variety of purposes.\n"
5             "Some common ones are:\n"
6             "\t\tto print double quotes \", use \\\"\\n"
7             "\t\tto print backslash \\", use \\\"\\n"
8             "\t\tto go to a new line, use \\n\\n"
9             "\t\tHowever to print the percentage sign %, we use %%%%\n");
10 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
C uses escape sequences for a variety of purposes.
Some common ones are:
•         to print double quotes ", use \"
          to print backslash \, use \\
          to go to a new line, use \n
          However to print the percentage sign %, we use %%
○ PS C:\Users\nehal\Downloads\University\Year 1\ENGG1410-Labs\Lab 1>
```

Part 2: Bill Splitting Program

Problem Statement

The problem for part two was to create a program that can help the user calculate the total bill by taking the price of the bill and adding the amount of tip to it, then splitting it between a certain number of people for even distribution.

Solution

To solve this problem, the user was first asked for the bill amount, the amount of tip, and how many people the bill has to be split into. Then the tip amount is calculated by multiplying bill amount by tip amount and dividing the whole answer by 100. Finally, the tip is added back into the original bill amount and divided equally between each person.

Conclusion and self assessment

We learned the importance of following the program requirements and then creating the program in an orderly and efficient fashion while ensuring all requirements are met.

Code

```
Lab 1 > C lab1bill.c > main(void)
1  #include <stdio.h>
2
3  int main(void){
4      //Variable decleration
5      float billAmount;
6      int tipPercentage;
7      int numPeople;
8
9      printf("Enter the original bill amount: ");
10     scanf("%f", &billAmount);
11     printf("Enter the tip percentage: ");
12     scanf("%d", &tipPercentage);
13     printf("Enter the number of people splitting the bill: ");
14     scanf("%d", &numPeople);
15
16     float tipAmount = (billAmount * tipPercentage) / 100;
17     float totalBill = billAmount + tipAmount;
18     float amountPerPerson = totalBill / numPeople;
19
20     printf("The total bill amount is: %.2f\n", totalBill);
21     printf("Each person should pay: %.2f\n", amountPerPerson);
22
23     return 0;
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

C uses escape sequences for a variety of purposes.
Some common ones are:

- to print double quotes ", use \"
- to print backslash \, use \\
- to go to a new line, use \n

However to print the percentage sign %, we use %%

© PS C:\Users\nehal\Downloads\University\Year 1\ENGG1410-Labs\Lab 1>

Part 3: Debugging

Problem Statement

The goal of this part was to debug a pre-written code and ensure that it runs the correct output without errors.

Solution

To solve this problem, the original syntax errors of the project were first fixed. Then the code was modified to find the amount of pounds by multiplying kilograms by pounds per kilogram. Then ounces were found by multiplying pounds by ounces per pound.

Conclusion and self assessment

We learned the importance of reviewing the code to ensure it runs as intended even if no errors are present, making sure to correct it if it doesn't run properly.

```
ENGG1410-Labs > Lab 1 > C lab1debug.c > ...
1  #include <stdio.h>
2
3  int main(void){
4  const double KgPerPound = 2.20;
5  const double OuncesPerPound = 16.0;
6  double weight;
7
8  printf("Please enter a weight in kilograms: ");
9  scanf("%lf", &weight);
10 double weightInPounds = weight * KgPerPound;
11
12 //Convert remaining pounds to ounces
13 double totalOunces = weightInPounds * OuncesPerPound;
14
15 printf("%.2lf kilograms, %.2lf pounds, %.2lf ounces\n", weight, weightInPounds, totalOunces);
16 return 0;
17 }
18
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Please enter a weight in kilograms: 10
10.00 kilograms, 22.00 pounds, 352.00 ounces
PS C:\Users\nehal\Downloads\University\Year 1\ENGG1410-Labs\Lab 1> |
```

Lab 2:

Part 1:

Problem Statement

The problem for this part of the lab was to understand how to run and use .json files and what their purposes were.

Solution

To understand the uses of .json files we had gone and explored the different uses of each of them and understanding their practical uses.

Conclusion and self assessment

We had learned the different practical applications of different .json files. The use of settings.json is used to help configure your personal user settings such as the appearance of the editor, editing default user configurations or even changing keyboard shortcuts. The use of tasks.json is to change the way the code is compiled like where it's printed to or how to manage "tasks". The last one was launch.json which helps customize how the debugging of the code should go or how the code should run.

Part 2:

Problem Statement

For part 2 we had to explore the different features of Vs code such as column selection, multi-cursor selection and more. Afterwards we had to use some of these features to edit a preset code.

Solution

The way we went around solving these problems was to use the new features in assisting in the use of VS code. After that to solve the problem we would take a look at the section we needed to edit or replace and talk amongst ourselves on how we could go around solving it. Once we figured out a course of action we would either edit the respective part or replace the part with a new section all together.

Conclusion and self assessment

We learned how to understand the different aspects of VS code as well as how we can manipulate code to give a different answer from before. We had learned how to work around syntax errors and how they can be prevented for the future.