UNIVERSITI TUN HUSSEIN ONN MALAYSIA

BIC10204: ALGORITHM & PROGRAMMING

SEMESTER 1, SESSION 2023/2024

PROJECT TITLE: TELECOM BILLING PROGRAM FOR KEBABO INTERNET SERVICES

GROUP A1

STUDENT'S NAME	MATRIX NO.
1. KHOR HWEE SHIAN	CI230085
2. YUVANESH A/L SHANMUGAM	CI230123
3. HAJIDAH BINTI AZIZI	CI230091
4. KEE JIA WEN	AI230134
5. LOW EURYCO	AI230155

Table of Content

1.0 Project Background

1.1 Title Description

The title suggests that the question revolves around creating a program for handling the billing system of Kebabo Internet Services, a fictional internet service provider. The focus is on calculating the monthly bill for customers based on their usage of weekday, night, and weekend minutes, considering specific pricing details and service taxes.

1.2 Theme Description

The theme involves simulating the billing process of a telecommunications company, specifically Kebabo Internet Services. It incorporates various elements such as a flat rate for weekday minutes, additional charges for exceeding allocated weekday minutes, free night and weekend minutes, and service taxes. The goal is to create a program that accurately computes the monthly bill for customers while adhering to the specified pricing structure.

1.3 Objective Description

1. User Input Handling. Prompt the user for the number of weekday minutes, night minutes, and weekend minutes used, ensuring entries are non-negative integers and handling potential errors appropriately.

2. Billing Calculation. Calculate the monthly bill by applying a flat rate for the first 600 weekday minutes, adding charges for any additional weekday minutes beyond 600 (RM0.20 each), and considering free night and weekend minutes.

3. Average Minute Cost Calculation. Calculate the average cost per minute before taxes by determining the total bill amount, calculating the total number of minutes used, and computing the average minute cost through division.

4. Service Tax Application. Apply a 5.5% service tax on all relevant charges, including the flat rate for the initial 600 weekday minutes and accounting for additional charges for weekday minutes exceeding 600.

5. Display results with clear labels and information. Show input data of number of weekday, night, and weekend minutes and present the amount before tax bill and average minute cost. Last, display the Service Tax (SST) cost separately and show the total bill.

2.0 Task Assignment

In managing the team's responsibilities, Khor Hwee Shian, the team leader, adeptly allocates tasks, ensuring an equitable and logical distribution among team members. The coding aspect, integral to project development, sees each team member assigned specific responsibilities aligned with their strengths and expertise. Simultaneously, the report writing component, crucial for documenting and communicating project details, is handled with equal diligence.

2.1 Group Member Roles in Coding Part

Khor Hwee Shian has a smart way of working. First, he makes the basic parts of the computer program, like the buttons and what you see on the screen. This makes it easier for the rest of the team to do their coding work. It helps everything run smoothly and saves time.

Yuvanesh A/L Shanmugam really understands how to solve the problem. He's good at writing code that figures out how much money to charge before taxes. He made a rule: if you use the internet a lot on weekdays and go over the basic time, you pay extra. But if you don't go over that time, you just pay the regular amount. This helps make sure people are charged the right amount for using the internet.

Hajidah Binti Azizi helps by adding a feature that figures out how much time you spent using the computer. She puts it in a smart setup that keeps checking until you get the info you need. This makes it easy for users to see how much they'll pay next time they use the internet. It makes the program more friendly and easy to use.

Kee Jia Wen does an important job by writing notes in the code. These notes help other coders understand what each part of the code does. It's like a guide that makes it easier for everyone to work together and understand the program better. Kee Jia Wen's work helps the team cooperate well.

Finally, Low Euryco makes the program look nice. This makes it more pleasant to use because it looks good. It helps people use the program more easily and enjoyably. Low Euryco's work adds a good finishing touch to the project.

2.2 Group Member Roles in Report Part

Khor Hwee Shian leads the team in creating the design report. They're in charge of important parts like the cover, table of contents, adjusting references, explaining the project's background, and task assignment. In the project background, Khor gives useful details about the program's title and theme, and they outline what the program aims to do. The task assignment part shows what each team member does in both coding and writing the report, revealing how everyone works together.

Yuvanesh A/L Shanmugam focuses on technical stuff, dealing with pseudocode, flowcharts, and presentation slides. He explains how the pseudocode and flowchart work, helping others understand the steps taken to solve the problem.

Hajidah Binti Azizi talks about what the program shows as output. She gives three examples of input and what the program would show as a result. This helps show how the program really works.

Kee Jia Wen talks about the program's code. He explains the different parts of the code that are important for finishing the project. He tells about the features in the C programming language used in the project and shows examples of the code. He also explains why the group chose to use those features, making it easier to understand their choices.

Finally, Low Euryco talks about the team's experience using C programming for the project. He discusses the challenges they faced, what they learned, and how well their approach worked. The conclusion gives ideas for making things better in the future, wrapping up the report with a full picture of the team's journey using C programming to solve the problem.

3.0 Pseudocode and Flowchart

In this section, we will explain the steps to create the program that calculates the monthly bill for Kebabo Internet Services. The pseudo-code and flowchart will be our guides. By the end of this section, we will understand how the program figures out the monthly bill.

3.1 Pseudocode

```
START
Set FLAT RATE as 39.99
Set FLAT MIN as 600
Set ADDICTIONAL COST PER MIN as 0.2
Set TAX as 0.055
Insert Name
Insert Number of Weekday Min
Insert Number of Night Min
Insert Number of Weekend Min
Read Name
Read Number of Weekday Min
Read Number of Night Min
Read Number of Weekend Min
If Number of Weekday Min is greater than FLAT MINUTE, then
    Compute the Pre Tax Bill as Pre Tax Bill = ((Weekday Min minus FLAT MIN) times by ADDITIONAL COST PER MIN) plus FLAT RATE
If Number of Weekday Min is less than FLAT MINUTE, then
    Compute the Pre Tax Bill as Pre Tax Bill = FLAT RATE
Compute the SST Cost as SST Cost = Pre Tax Bill times TAX
Compute Average Min Cost as Average Minute Cost = ((Weekday Min plus Night Min plus Weekend Min) minus FLAT MIN) divide by (FLAT RATE times 100)
Compute the Total Bill as Total Bill = Pre Tax Bill plus SST COST
Display the Name
Display Number of Weekday Min
Display Number of Night Min
Display Number of Weekend Min
Display the Average Min Cost
Display the SST Cost
Display the Pre Tax Bill
Display the Total Bill
END
```

3.2 Flowchart

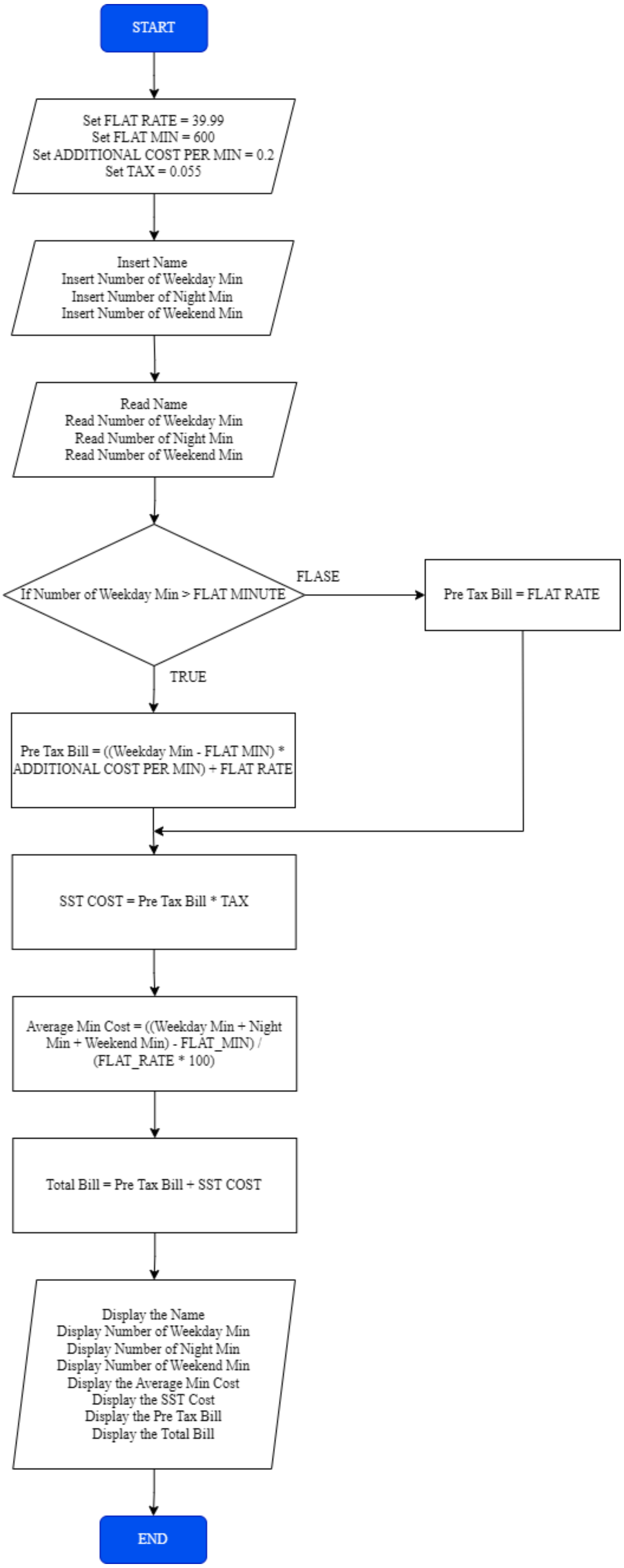
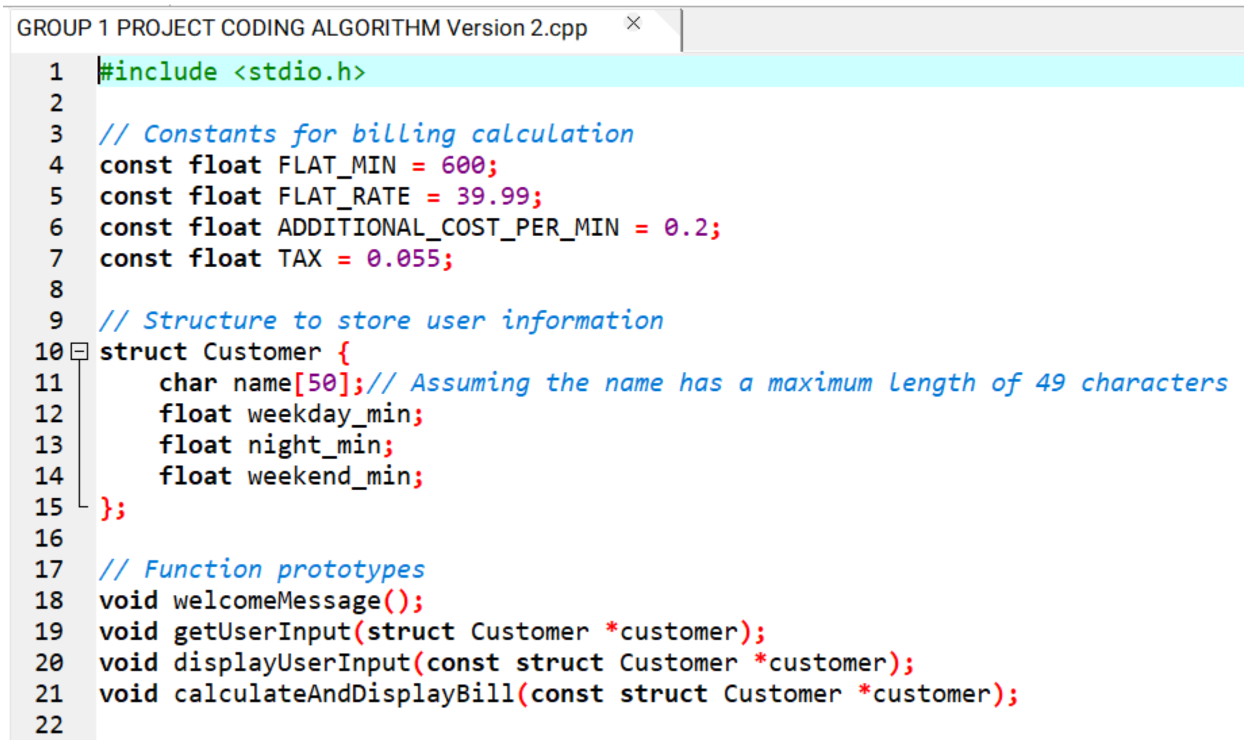

START
Set FLAT RATE = 39.99
Set FLAT MIN = 600
Set ADDITIONAL COST PER MIN = 0.2
Set TAX = 0.055
Insert Name
Insert Number of Weekday Min
Insert Number of Night Min
Insert Number of Weekend Min
Read Name
Read Number of Weekday Min
Read Number of Night Min
Read Number of Weekend Min
If Number of Weekday Min > FLAT MINUTE
FLASE
Pre Tax Bill = FLAT RATE
TRUE
Pre Tax Bill = ((Weekday Min - FLAT MIN) * ADDITIONAL COST PER MIN) + FLAT RATE
SST COST = Pre Tax Bill * TAX
Average Min Cost = ((Weekday Min + Night Min + Weekend Min) - FLAT_MIN) / (FLAT_RATE * 100)
Total Bill = Pre Tax Bill + SST COST
Display the Name
Display Number of Weekday Min
Display Number of Night Min
Display Number of Weekend Min
Display the Average Min Cost
Display the SST Cost
Display the Pre Tax Bill
Display the Total Bill
END

Figure 1 Flowchart of Telecom Billing Program for Kebabo Internet Services

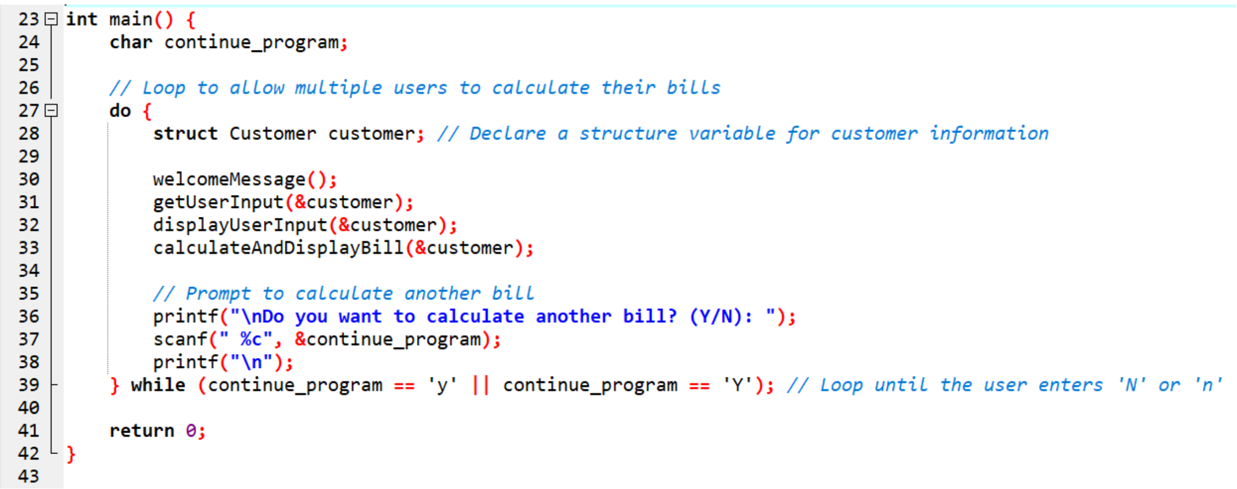
4.0 Program Code Description



```
1  #include <stdio.h>
2
3  // Constants for billing calculation
4  const float FLAT_MIN = 600;
5  const float FLAT_RATE = 39.99;
6  const float ADDITIONAL_COST_PER_MIN = 0.2;
7  const float TAX = 0.055;
8
9  // Structure to store user information
10 struct Customer {
11     char name[50];// Assuming the name has a maximum length of 49 characters
12     float weekday_min;
13     float night_min;
14     float weekend_min;
15 };
16
17 // Function prototypes
18 void welcomeMessage();
19 void getUserInput(struct Customer *customer);
20 void displayUserInput(const struct Customer *customer);
21 void calculateAndDisplayBill(const struct Customer *customer);
22
```

Figure 2 Explanation of line 1 to line 22

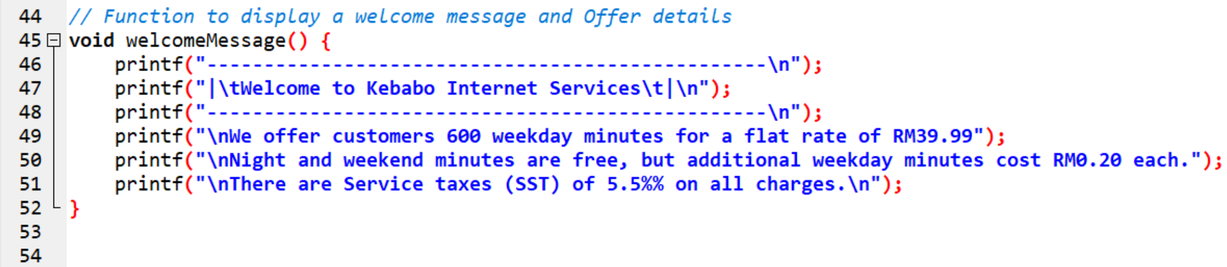
The program defines variables to store billing information, including the total bill, pre-tax bill, sst cost, average cost per minute, flat minute allowance, flat rate, additional cost per minute, and service tax rate. In first line, we include standard input and output file into source program which has the information for all input, output related functions. In line 3 to line 7, variables related to the internet service billing information such as the flat minute allowance, flat rate, additional cost per minute and service tax rate are assigned value. The flat minute allowance provided to customers is 600 minutes. The flat rate for the provided flat minute allowance is RM 39.99. Additional cost per minute beyond the flat rate is RM 0.20. Last, the service tax is 5.5%. In line 10 to line 15, there are a structure to store user information which are char data type name identifier, float data type weekday min identifier, float data type night min identifier and float data type weekend min identifier. In line 18 to line 21, we call function such as welcome message function and three function named get user input function, display user input function, calculate and display bill function, these function are contain customer's indirection pointer operator.



```
23  int main() {
24      char continue_program;
25  
26      // Loop to allow multiple users to calculate their bills
27      do {
28          struct Customer customer; // Declare a structure variable for customer information
29  
30          welcomeMessage();
31          getUserInput(&customer);
32          displayUserInput(&customer);
33          calculateAndDisplayBill(&customer);
34  
35          // Prompt to calculate another bill
36          printf("\nDo you want to calculate another bill? (Y/N): ");
37          scanf(" %c", &continue_program);
38          printf("\n");
39      } while (continue_program == 'y' || continue_program == 'Y'); // Loop until the user enters 'N' or 'n'
40  
41      return 0;
42  }
43  
```

Figure 3 Explaination of line 23 to line 43

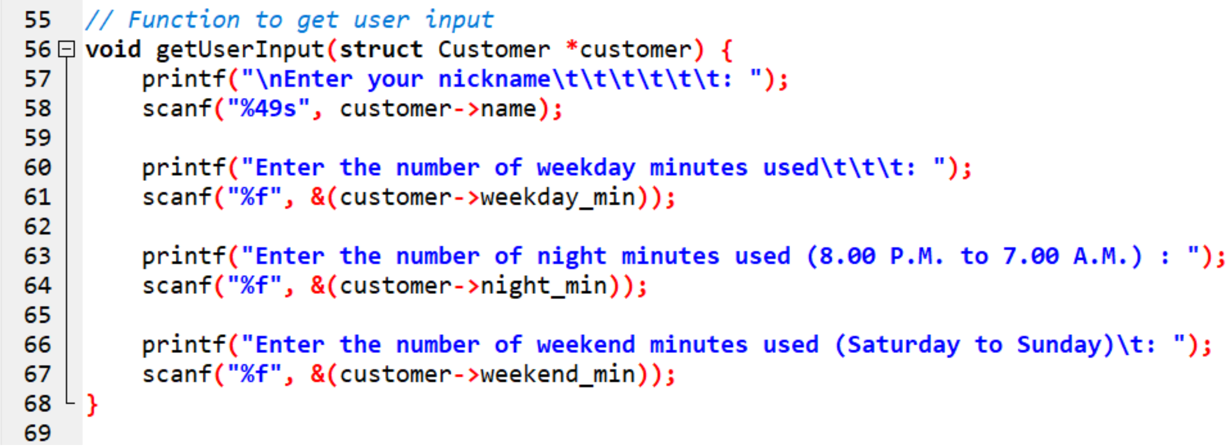
The main function includes do while loop structure to allow user to insert multiple bill calculations. In line 24, there are char data type and continue_program identifier to make user can repeat the program again and again until user type ‘N’. In line 28, it is a struct to store user’s all information in this time program. Next, line 30 to line 33 is about four function which are welcome message function and three function named get user input function, display user input function, calculate and display bill function with customer’s address pointer operator,which mean it can copy and paste the user’s information. The program prompts the user to input their nickname and the number of weekday minutes used, number of night minutes used and number of weekend minutes used. Welcome messages and offer details about Internet service plan such as the user's nickname and the minutes used during different times of the day (weekday_min, night_min, weekend_min) are displayed. The program will calls next function and passes the user's input (weekday minutes, weekend minutes, and night minutes) as arguments to the next function. The program then displays billing information for users of a hypothetical internet service provider called "Kebabo Internet Services." A do-while loop asks the users if they want to calculate another bill. If users enter ‘y’, program allows users to calculate bills for multiple customers. The loop continues until the user chooses not to calculate another bill, which is when users enter any other characther than ‘y’ or ‘Y’. The main function returns 0, indicating successful program execution.



```
44  // Function to display a welcome message and Offer details
45  void welcomeMessage() {
46      printf("------------------------------------------------\n");
47      printf("|\tWelcome to Kebabo Internet Services\t|\n");
48      printf("------------------------------------------------\n");
49      printf("\nWe offer customers 600 weekday minutes for a flat rate of RM39.99");
50      printf("\nNight and weekend minutes are free, but additional weekday minutes cost RM0.20 each.");
51      printf("\nThere are Service taxes (SST) of 5.5%% on all charges.\n");
52  }
53  
54  
```

Figure 4 Explaination of line 44 to line 54

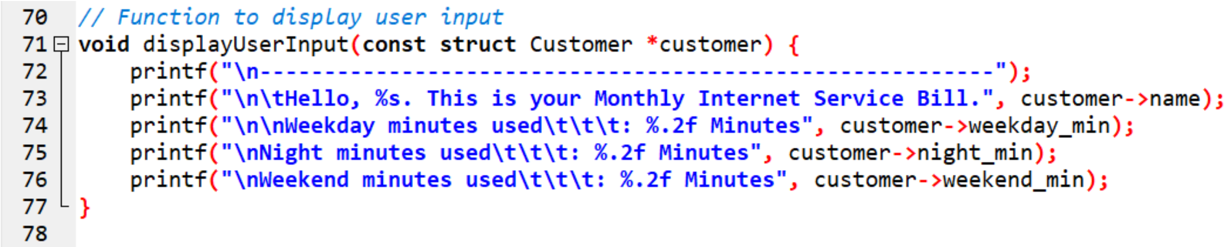
The welcome message function is designed to smoothly greet users while providing detailed information about an offer. Its main goal is to make the user experience friendly, reducing the chance of missing important details by making it easy for users to understand and remember the information presented.



```
55    // Function to get user input
56 ⊟ void getUserInput(struct Customer *customer) {
57        printf("\nEnter your nickname\t\t\t\t\t\t: ");
58        scanf("%49s", customer->name);
59
60        printf("Enter the number of weekday minutes used\t\t\t: ");
61        scanf("%f", &(customer->weekday_min));
62
63        printf("Enter the number of night minutes used (8.00 P.M. to 7.00 A.M.) : ");
64        scanf("%f", &(customer->night_min));
65
66        printf("Enter the number of weekend minutes used (Saturday to Sunday)\t: ");
67        scanf("%f", &(customer->weekend_min));
68    }
69
```

Figure 5 Explaination of line 55 to line 69

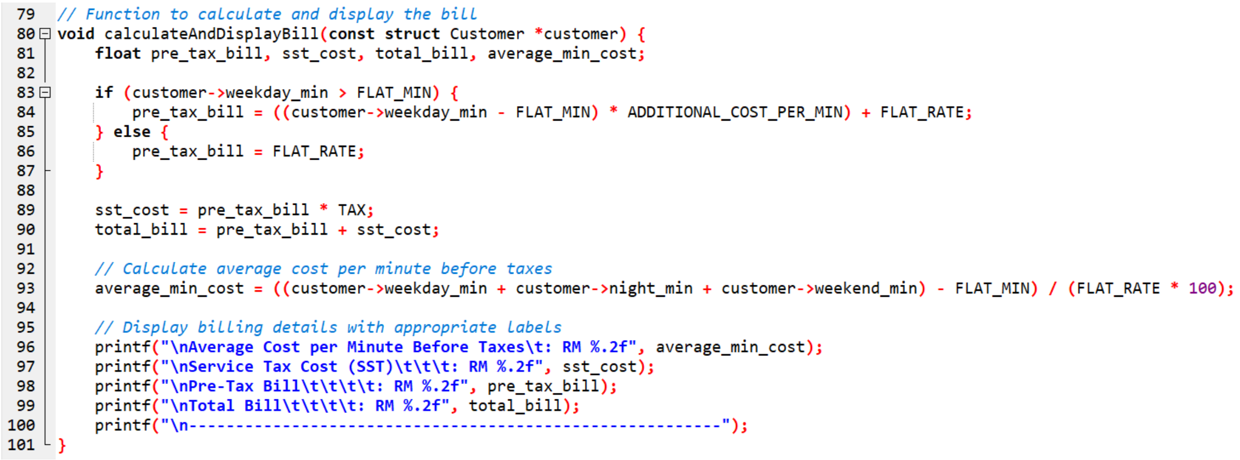
The purpose of the get user input function is to collect and save important information from users. This includes their name and the number of minutes they spend on weekdays, during the night (from 8 PM to 7 AM), and on weekends (from Saturday to Sunday). The function is designed to handle these details carefully, taking into account specific time periods for night and weekend minutes.



```
70    // Function to display user input
71 ⊟ void displayUserInput(const struct Customer *customer) {
72        printf("\n---------------------------------------------------------");
73        printf("\n\tHello, %s. This is your Monthly Internet Service Bill.", customer->name);
74        printf("\n\nWeekday minutes used\t\t\t: %.2f Minutes", customer->weekday_min);
75        printf("\nNight minutes used\t\t\t: %.2f Minutes", customer->night_min);
76        printf("\nWeekend minutes used\t\t\t: %.2f Minutes", customer->weekend_min);
77    }
78
```

Figure 6 Explaination of line 70 to line 78

The display user input function is like a show-and-tell for the weekday, night, and weekend minutes a customer has used. First, the program calls the customer's name to let them know that the billing process is finished. After that, it shows the customer weekday, night and weekend minutes used they just entered into the system, making sure everything is clear and easy to understand.



```c
 79  // Function to calculate and display the bill
 80  void calculateAndDisplayBill(const struct Customer *customer) {
 81      float pre_tax_bill, sst_cost, total_bill, average_min_cost;
 82  
 83      if (customer->weekday_min > FLAT_MIN) {
 84          pre_tax_bill = ((customer->weekday_min - FLAT_MIN) * ADDITIONAL_COST_PER_MIN) + FLAT_RATE;
 85      } else {
 86          pre_tax_bill = FLAT_RATE;
 87      }
 88  
 89      sst_cost = pre_tax_bill * TAX;
 90      total_bill = pre_tax_bill + sst_cost;
 91  
 92      // Calculate average cost per minute before taxes
 93      average_min_cost = ((customer->weekday_min + customer->night_min + customer->weekend_min) - FLAT_MIN) / (FLAT_RATE * 100);
 94  
 95      // Display billing details with appropriate labels
 96      printf("\nAverage Cost per Minute Before Taxes\t: RM %.2f", average_min_cost);
 97      printf("\nService Tax Cost (SST)\t\t\t: RM %.2f", sst_cost);
 98      printf("\nPre-Tax Bill\t\t\t\t: RM %.2f", pre_tax_bill);
 99      printf("\nTotal Bill\t\t\t\t: RM %.2f", total_bill);
100      printf("\n---------------------------------------------------------");
101  }
```

Figure 7 Explaination of line 79 to line 101

This code defines a function named calculate. The function takes three parameters which include weekday_min, weekend_min, and night_min, representing the number of minutes used during weekdays, weekends, and nights, respectively. The code checks if the weekday minutes exceed the flat rate. If yes, it calculates the pre tax bill as additional cost for exceeding minutes and adds the flat rate. If not, it assigns the pre-tax bill directly, which is RM39.99, same as the flat rate. Service tax cost is calculated by multiplying the pre-tax bill by the service tax rate. The total bill is calculated by adding the pre-tax bill and the service tax cost. The average cost is the pre-tax bill divided by the total minutes used during weekend, weekday and night. The function then prints various billing details, including the pre-tax bill, sst cost, total bill, and average cost per minute. The output is displayed with two decimal places in Malaysian Ringgit (RM).

5.0 Output Description

5.1 Introduction

Input code are showing the code is a billing calculation program for an internet service provider named Kebabo Internet Services. It calculates a customer's internet service bill based on their usage in minutes and the service's pricing structure. The pricing structure consists of a flat rate for a set allowance of minutes and an additional charge for minutes used beyond this allowance. The key billing parameters defined in the program are as follows:

1. Flat Minute Allowance: The allowance is 600 weekday minutes, as denoted by the FLAT_MIN constant.

2. Flat Rate Cost: The flat cost for these 600 minutes is set at 39.99 (currency not specified), indicated by the FLAT_RATE constant.

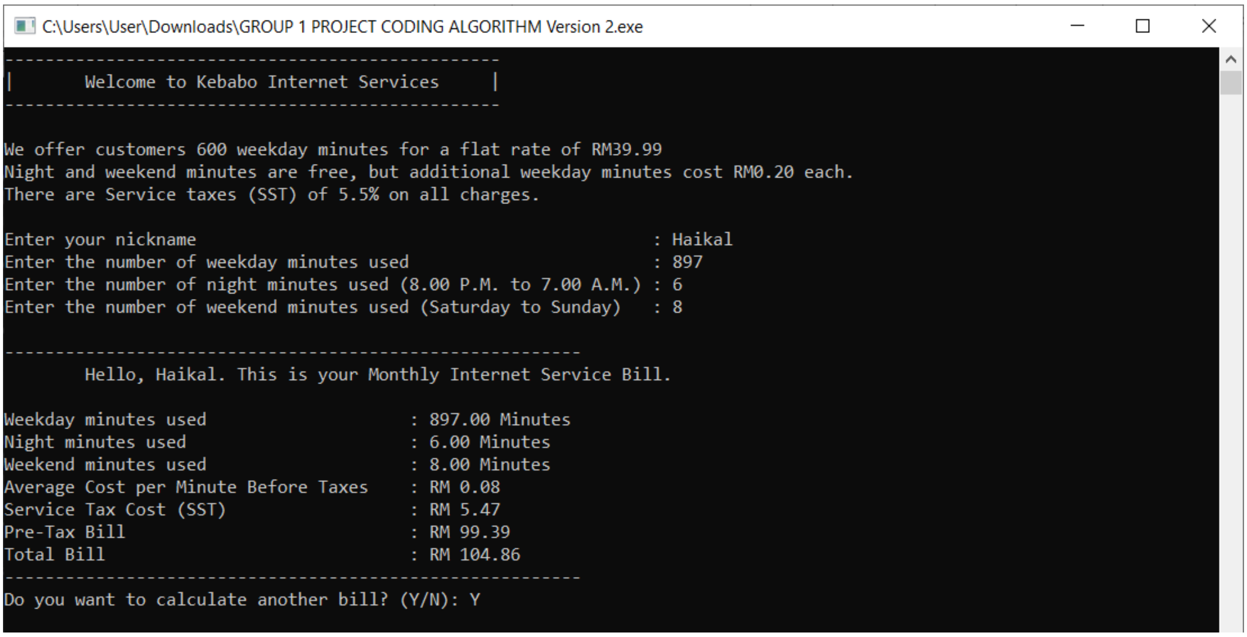
3. Additional Cost Per Minute: If a customer exceeds the 600-minute allowance during weekdays, there is an additional cost of 0.20 per minute, given by the ADDITIONAL_COST_PER_MIN constant.

4. Service Tax (SST): On top of the pre-tax bill, a Service Sales Tax (SST) of 5.5% is applied, defined by the TAX constant.

The program's flow begins by prompting the user to enter their nickname and the minutes used during weekdays, nights, and weekends. It then employs a function called calculate to determine the total pre-tax bill based on the minutes used. The calculate function applies the flat rate if the weekday usage is within the allowance. If the usage exceeds the allowance, it calculates the additional cost for the extra minutes used. After computing the pre-tax bill, the function calculates the SST and adds it to the pre-tax bill to arrive at the total bill amount.

Additionally, the program calculates and displays the average cost per minute before taxes. This is done by dividing the total pre-tax bill by the total minutes used. Finally, the program displays a detailed bill that includes the average cost per minute, SST amount, pre-tax bill, and total bill. It also allows the user to run multiple calculations by looping back to the start if the user wishes to calculate another bill. The program is designed to be user-friendly, providing clear prompts and displaying the bill with a detailed breakdown.

5.2 Haikal's Monthly Internet Service Bill



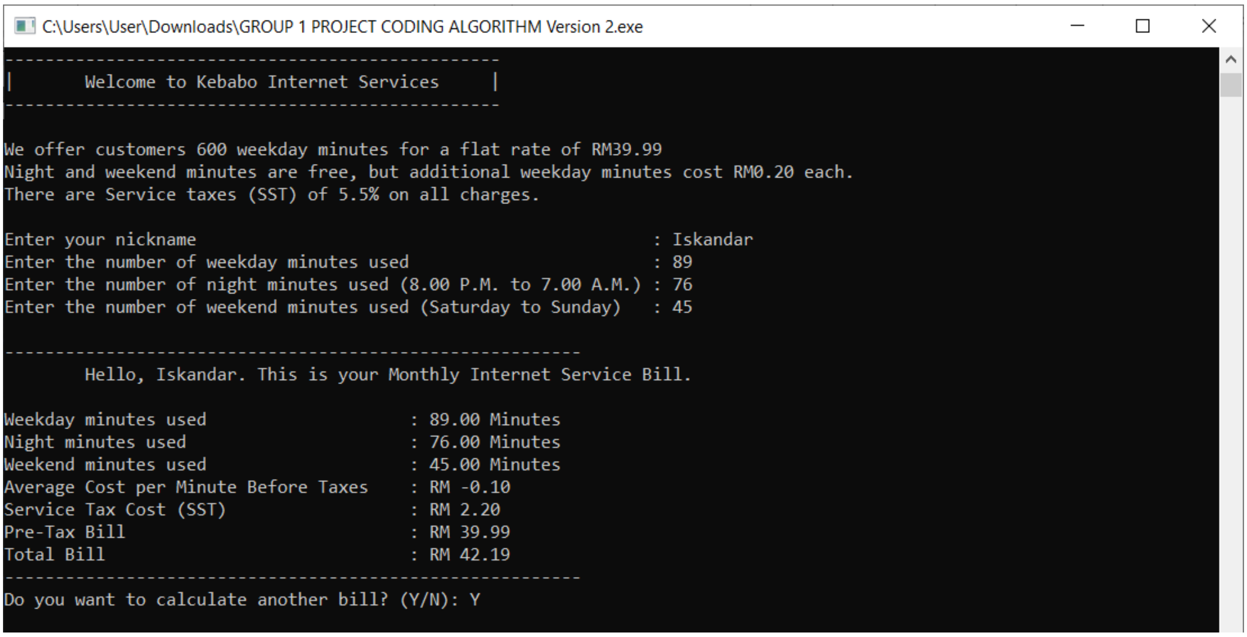
C:\Users\User\Downloads\GROUP 1 PROJECT CODING ALGORITHM Version 2.exe

```
-------------------------------------------------
|       Welcome to Kebabo Internet Services     |
-------------------------------------------------

We offer customers 600 weekday minutes for a flat rate of RM39.99
Night and weekend minutes are free, but additional weekday minutes cost RM0.20 each.
There are Service taxes (SST) of 5.5% on all charges.

Enter your nickname                                     : Haikal
Enter the number of weekday minutes used                : 897
Enter the number of night minutes used (8.00 P.M. to 7.00 A.M.) : 6
Enter the number of weekend minutes used (Saturday to Sunday)   : 8

---------------------------------------------------------
        Hello, Haikal. This is your Monthly Internet Service Bill.

Weekday minutes used                    : 897.00 Minutes
Night minutes used                      : 6.00 Minutes
Weekend minutes used                    : 8.00 Minutes
Average Cost per Minute Before Taxes    : RM 0.08
Service Tax Cost (SST)                  : RM 5.47
Pre-Tax Bill                            : RM 99.39
Total Bill                              : RM 104.86
---------------------------------------------------------
Do you want to calculate another bill? (Y/N): Y
```

Figure 8 Haikal's Monthly Internet Service Bill

For Haikal monthly internet service bill in figure 2, the system records 897 weekday minutes, 6 night minutes, and 8 weekend minutes. Despite the significant usage during the weekdays, the total due is RM 104.86, reflecting the flat rate plus a service tax of 5.5%, which amounts to RM 5.47. This suggests that the weekday minutes fall within a flat-rate plan, while the night and weekend minutes, which are less, do not contribute additional charges.

5.3 Iskandar's Monthly Internet Service Bill



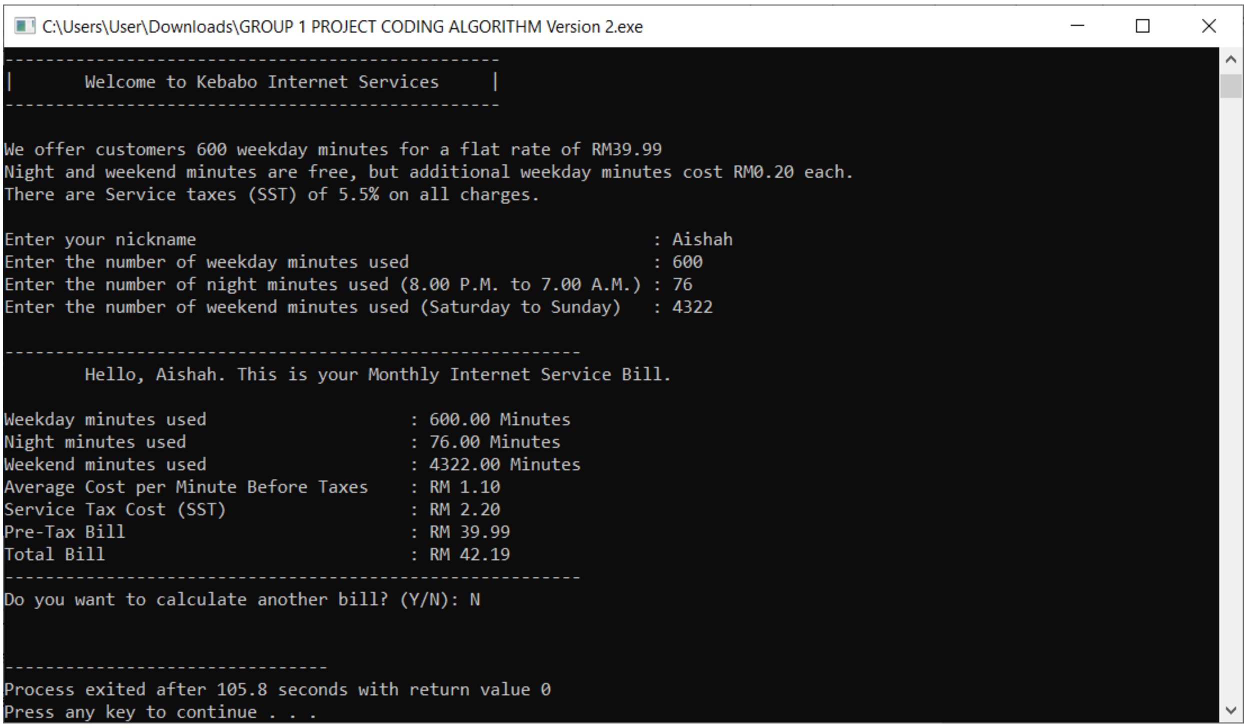
C:\Users\User\Downloads\GROUP 1 PROJECT CODING ALGORITHM Version 2.exe

```
------------------------------------------------
|       Welcome to Kebabo Internet Services     |
------------------------------------------------

We offer customers 600 weekday minutes for a flat rate of RM39.99
Night and weekend minutes are free, but additional weekday minutes cost RM0.20 each.
There are Service taxes (SST) of 5.5% on all charges.

Enter your nickname                                     : Iskandar
Enter the number of weekday minutes used                : 89
Enter the number of night minutes used (8.00 P.M. to 7.00 A.M.) : 76
Enter the number of weekend minutes used (Saturday to Sunday)   : 45

--------------------------------------------------------
        Hello, Iskandar. This is your Monthly Internet Service Bill.

Weekday minutes used                    : 89.00 Minutes
Night minutes used                      : 76.00 Minutes
Weekend minutes used                    : 45.00 Minutes
Average Cost per Minute Before Taxes    : RM -0.10
Service Tax Cost (SST)                  : RM 2.20
Pre-Tax Bill                            : RM 39.99
Total Bill                              : RM 42.19
--------------------------------------------------------
Do you want to calculate another bill? (Y/N): Y
```

Figure 9 Iskandar's Monthly Internet Service Bill

Moving to Iskandar's monthly internet service bill in figure 3, the system shows lower usage across the board with 89 weekday minutes, 76 night minutes, and 45 weekend minutes. There's an unusual negative average cost per minute before taxes, which could indicate a rebate or credit applied to the account or a flaw in the system's calculation logic. The total bill is RM 42.19, including a service tax of RM 2.20, aligning with the flat-rate plan despite the negative average cost per minute suggesting some form of discount or adjustment has been made.

5.4 Aishah's Monthly Internet Service Bill


C:\Users\User\Downloads\GROUP 1 PROJECT CODING ALGORITHM Version 2.exe

```
-------------------------------------------------
|       Welcome to Kebabo Internet Services     |
-------------------------------------------------

We offer customers 600 weekday minutes for a flat rate of RM39.99
Night and weekend minutes are free, but additional weekday minutes cost RM0.20 each.
There are Service taxes (SST) of 5.5% on all charges.

Enter your nickname                                      : Aishah
Enter the number of weekday minutes used                 : 600
Enter the number of night minutes used (8.00 P.M. to 7.00 A.M.) : 76
Enter the number of weekend minutes used (Saturday to Sunday)   : 4322

---------------------------------------------------------
        Hello, Aishah. This is your Monthly Internet Service Bill.

Weekday minutes used                : 600.00 Minutes
Night minutes used                  : 76.00 Minutes
Weekend minutes used                : 4322.00 Minutes
Average Cost per Minute Before Taxes    : RM 1.10
Service Tax Cost (SST)              : RM 2.20
Pre-Tax Bill                        : RM 39.99
Total Bill                          : RM 42.19
---------------------------------------------------------
Do you want to calculate another bill? (Y/N): N


--------------------------------
Process exited after 105.8 seconds with return value 0
Press any key to continue . . .
```

Figure 10 Aisyah's Monthly Internet Service Bill

Aishah's monthly internet service bill in figure 4, however, presents an anomaly. While the weekday and night minutes used are similar to Haikal's, the weekend minutes are excessively high at 4322. Under a typical billing system, such an exorbitant number of minutes would result in a substantial charge, but the bill totals to RM 42.19, which is surprisingly the same as Iskandar's and includes a service tax of RM 2.20. This raises questions about the accuracy of the weekend minute calculations or if there's a cap on the charges irrespective of the minutes used.

5.5 Conclusion

The images depict outputs from what appears to be a command-line interface for a telecommunications billing system, which calculates the cost of phone usage for customers. Each output represents a different customer's bill, taking into account their use of minutes during various times of the week.

In all three cases, the pre-tax bill reflects a flat rate of RM 39.99, indicating that the primary variable affecting the total cost is the service tax based on the pre-tax amount. The consistency of the pre-tax bill suggests a plan structure that offers a certain number of minutes at a fixed cost, with the service tax being the only variable. Each output concludes with a prompt for the user to calculate another bill, which is a common feature in interactive applications to allow for continuous operations. This billing system seems to have a straightforward approach to charges for minute usage, but the presented data points to possible inconsistencies in the application of charges for additional minutes or potential errors in the billing calculations, especially for weekend minutes, which should be investigated further. Then when user put N to the part to perform the another bill , the system will stop the program.

6.0 Discussion and Conclusion

6.1 Discussion of the Program

The provided C program effectively implements the logic to calculate the monthly bill for Kebabo Internet Services based on the user's input of weekday, night, and weekend minutes. It incorporates the flat rate for weekday minutes, additional charges for exceeding the 600 weekday minutes limit, and taxes (SST) on all charges.

The program uses clear and concise variable names and follows good coding practices, making it easy to understand and maintain. It employs appropriate data types, such as integers for minute counts and floating-point numbers for currency values.

One notable aspect of the program is its user-centric design. The inclusion of prompts for users to input the number of weekday, night, and weekend minutes ensures an interactive and user-friendly experience. This not only contributes to the overall usability of the program but also aligns with modern software design principles that prioritize user engagement and ease of use.

The input is taken from the user through the standard input (stdin), and the output is displayed with meaningful labels. The rounding of the costs to whole cents is crucial for accurate financial representation. By rounding the taxes, average minute cost, and total bill to two decimal places, the program ensures a realistic and precise reflection of the financial calculations.

The output presentation of the program is another commendable feature. The results are displayed with meaningful labels, offering users a clear breakdown of the billing components. The decision to round costs to whole cents is a subtle yet crucial detail, reflecting an understanding of real-world financial practices. This attention to detail ensures that the financial representation in the program aligns with standard billing conventions.

The comments in the code enhances readability and provides insights into the purpose of each section. It helps both developers and future maintainers understand the logic and calculations performed at each step.

Looking ahead, potential enhancements to the program could involve the integration of additional error-checking mechanisms to handle invalid inputs gracefully. While the current implementation assumes valid user inputs, incorporating robust error handling would further bolster the program's reliability and resilience in real-world usage.

6.2 Conclusion of the Program

In conclusion, the C program successfully addresses the requirements outlined for calculating the monthly bill for Kebabo Internet Services. It effectively takes user input, performs calculations for the pre-tax bill, average minute cost, SST cost, and total bill, and presents the results in a clear and organized manner. The use of constants and appropriate variable naming enhances the program's readability, making it easy to understand and maintain.

The program adopt to the specified billing structure, accurately accounting for flat rates, additional charges for exceeding limits, and the application of service taxes. The inclusion of rounding ensures that all monetary values are stored as whole cents, aligning with standard billing practices. Overall, the program serves as a functional tool for users to estimate and understand their monthly internet service costs, incorporating key elements such as usage patterns and associated charges.

7.0 References

[1] Heidihale58. (2023, April). Pseudocode for BMI. stackoverflow. https://stackoverflow.com/questions/76142464/pseudocode-for-bmi.

[2] None. (None). If-else Flowchart. Zen Flowchart. https://www.zenflowchart.com/guides/if-else-flowchart.

[3] None. (None). Chapter 10- Pseudocode and Flowcharts. Computer Science IGCSE. https://anatcomputerscience.wordpress.com/chapter-10/.

[4] Desmond Motiso. (2022, December). What Is Source Code in Programming? (Definition and Example). Indeed. https://www.indeed.com/career-advice/career-development/what-is-source-code.

[5] Rathoadavinash. (2023, April). Introduction to the C99 Programming Language Part I.Geeksforgeeks. https://www.geeksforgeeks.org/introduction-to-the-c99-programming-language-part-i/?ref=gcse.

[6] Dixit, J.B. (2005). Fundamentals of computers and programming in C. Internet Archive. https://archive.org/details/fundamentalsofco0000dixi/page/n3/mode/2up.

[7] AdarshKumarSingh. (None). Customer Billing System Project Using C Language. Study Tonight. https://www.studytonight.com/c-projects/customer-billing-system-project-using-c-language.

[8] Kunal Metha. (2022, January 5). C if...else Statement. SCALER Topics. https://www.scaler.com/topics/c/if-else-statement-in-c/.

[9] Barbara Thompson. (2023, December 11). Loops in C: For, While, Do While looping Statements. guru99. https://www.guru99.com/c-loop-statement.html.