

Bangladesh University of Business & Technology (BUBT)



Lab Report

Course Title : Structured Programming Language Lab
Course Code : CSE 102

Submitted by

Name : Makhzum-Bin-Harun
ID : 20254103279
Intake : 56
Section : 07
Program: B.Sc. Engg. in CSE

Submitted to

Name : Sourav Kundu
Designation: Lecturer
Department of Computer
Science & Engineering
Bangladesh University of
Business & Technology.

Date of Submission: August 20, 2025

Signature of Teacher

Table of Contents

Problem 01.....	2
Problem 02.....	4
Problem 03.....	6

Problem 01. Write a C program to show the below text:

My name is Mr. Doe,
I am from USA,
My employee id : 4263

01.1. Algorithm: Algorithm to Print three lines given in the following problem

1. Start
2. Include the header file `<stdio.h>`
3. Define the `main()` function
4. Use `printf()` to display:

My name is Mr. Doe,

I am from USA,

My employee id: 4263.

5. End the program with `return 0;`
6. Stop

01.2. C Program Code:

```
#include <stdio.h>

int main() {
    printf("My name is Mr. Doe,\n");
    printf("I am from USA,\n");
    printf("My employee id: 4263.\n");
    return 0;
}
```

01.3. Output:

Output

Clear

My name is Mr. Doe,
I am from USA,
My employee id: 4263.

=== Code Execution Successful ===

01.4. Discussion:

In this program, we display text on the screen using the `printf()` function in C. The header file `<stdio.h>` is required since it contains the definition of input-output functions. The program execution

begins with the `main()` function. Each line of text is printed with the help of `printf()`, and the escape sequence `\n` is used to move the cursor to the next line so that each statement appears in a separate line. Finally, `return 0;` indicates that the program has executed successfully.

Problem 02. Suppose you have 500 Tk, you have to buy a product cost of 350.80 Tk, and later you got 200 Tk from your brother.

Write a C program where user can input the balance and finally can see how much money remain?

02.1. Algorithm: Algorithm to find remaining balance

1. Start
2. Declare three variables: balance, product_cost, and money_received (type float).
3. Ask the user to input the initial balance.
4. Ask the user to input the product cost.
5. Ask the user to input the amount received from the brother.
6. Calculate the remaining balance using the formula:

$$\text{balance} = \text{balance} - \text{product_cost} + \text{money_received}$$

7. Display the remaining balance (with 2 decimal places).
8. Stop

02.2. C Program Code:

```
#include <stdio.h>

int main() {
    float balance, product_cost, money_received;
    printf("Enter your initial balance: ");
    scanf("%f", &balance);
    printf("Enter the product cost: ");
    scanf("%f", &product_cost);
    printf("Enter the amount you received from your brother: ");
    scanf("%f", &money_received);
    balance = (balance-product_cost+money_received);
    printf("Remaining Balance: %.2f Tk\n", balance);

    return 0;
}
```

02.3. Output:

Output Clear

```
Enter your initial balance: 500
Enter the product cost: 350.80
Enter the amount you received from your brother: 200
Remaining Balance: 349.20 Tk

=== Code Execution Successful ===
```

02.4. Discussion:

In this program, we calculate the remaining balance after a purchase and receiving extra money. We first declare three floating-point variables to store the user's initial balance, product cost, and money received. Floating-point (float) is used because the values may include decimals (e.g., 350.80).

The program uses `scanf()` to take user inputs and `printf()` to display results. The calculation is performed using the formula:

$$\text{Remaining Balance} = \text{Initial Balance} - \text{Product Cost} + \text{Money Received}$$

Finally, the result is printed using `%.2f`, which ensures that the balance is shown up to two decimal places.

For example, if the inputs are:

Enter your initial balance: 500

Enter the product cost: 350.80

Enter the amount you received from your brother: 200

The calculation becomes:

Remaining Balance: 349.20 Tk

Thus, the program works correctly for different inputs and gives the remaining balance.

Problem 03. Write a C program to find if a number is positive or not?

Take number as input from keyboard and show output as below:

if positive:

5 is a positive number.

If not:

-5 is not a positive number.

03.1. Algorithm: Algorithm to Convert Days into Years, Weeks, and Days

1. Start
2. Declare an integer variable num.
3. Prompt the user to enter a number.
4. Take the number as input using scanf().
5. Check the condition:
 - If num > 0, print "num is a Positive Number."
 - Else if num == 0, print "num is neither Positive nor Negative."
 - Else (when num < 0), print "num is not a Positive Number."
6. Stop

03.2. C Program Code:

```
#include <stdio.h>

int main() {
    int num;
    printf("Enter a Number: ");
    scanf("%d", &num);

    if (num > 0) {
        printf("%d is a Positive Number.\n", num);
    }
    else if (num == 0) {
        printf("%d is neither Positive nor Negative.\n", num);
    }
    else {
        printf("%d is not a Positive Number.\n", num);
    }

    return 0;
}
```

03.3. Output:

Case 01:

Output Clear

```
Enter a Number: 5
5 is a Positive Number.

=== Code Execution Successful ===
```

Case 02:

Output Clear

```
Enter a Number: -5
-5 is not a Positive Number.

=== Code Execution Successful ===
```

Case 03:

Output Clear

```
Enter a Number: 0
0 is neither Positive nor Negative.

=== Code Execution Successful ===
```

03.4. Discussion:

In this program, we check whether a given number is positive or not. The program starts by declaring an integer variable `num` to store the input value. Using `scanf()`, the user provides a number.

- If the number is greater than 0, it is considered positive, and the program prints:

```
num is a Positive Number.
```

- If the number is exactly 0, it is neither positive nor negative, so the program prints:

```
0 is neither Positive nor Negative.
```

- If the number is less than 0, it is not positive, and the program prints:

num is not a Positive Number.

The use of if-else-if conditions ensures that only one correct output is printed depending on the input.