

## **Programming Fundamentals FALL - 2024**

### **ASSIGNMENT – 3 – SE**

**Deadline: Sunday Nov 24, 2024**

1. Make sure that you read and understand each instruction. If you have any questions or comments you are encouraged to discuss your problems with your colleagues (and instructors) on google classroom.
2. If there is a syntax error in the code, zero marks will be awarded in that part of the assignment.
3. Keep a backup of your work always that will be helpful in preventing any mishap and avoid last hour submissions
4. Displayed output should be well mannered and well presented. Use appropriate comments and indentation in your source code.
5. Please ensure that only concepts covered in class are used for the assignment. For each question in your assignment, make a separate .cpp file e.g., for question 1, make q1.cpp and so on. Each file that you submit must contain your name, student-id, and assignment on top of the file in the comments.
7. Combine all your work in one folder. The folder must contain only .cpp files (no binaries, no exe files etc.).
8. **Rename the zip as ROLL-NUM\_PROGRAM\_SECTION (e.g., i240001\_SE\_A) and compress the folder as a zip file. (e.g., i240001\_SE\_A.zip). Strictly follow this naming convention, otherwise marks will be deducted (-10%).**
9. Submit the .zip file on Google Classroom within the deadline.
10. The student is solely responsible for checking the final file for issues like corrupt file, viruses in the file, or mistakenly exe sent. If we cannot download the file from Google Classroom, it will lead to zero marks in the assignment.

**Note: Start early so that you can finish it on time.**

## **Question 1 - String Manipulation**

Design a String Manipulation Program that can be used to perform the following functions related to strings. **You cannot use any build in string function else zero marks will be awarded:**

### **String Manipulation Structure:**

The structure should have followed functions:

#### **1. int Calculate\_length (String s)**

The function should get a string as input and return the length of the string.

Example s = "Bilal", return will be 5;

#### **2. bool substring (String main, String str)**

Write code to identify the substring from any given string. The String main represent the actual string and String str represents the string that they should find e.g.

Main = "I am taking the OOP Class"

Str = "OOP"

The function should return true as the OOP exist in the above string.

Another example

Main = "I am taking the OOP Class"

Str = "taking"

The function should return true as it exists in the above string.

Another example

Main = "I am taking the OOP Class"

Str = "that OOP"

The function should return false as it does not exist in the above string.

Another example

Main = "I am taking the OOP Class"

Str = "Taking the oop class and doing the assignment"

The function should return false as it does not exist in the above string.

#### **3. int substring\_position (String main, String str)**

The above function returns the index of the main string where the sub string starts

For example

Main = "I am taking OOP course"

Str = "taking"

The function will return 5 as substring exist and starts at 5<sup>th</sup> index of main string. In case the substring does not exist, you will return -1.

### **Main function:**

- Ask the user for mains string.
- Make a menu using switch statement above three functions, call respective function and display output

**Please ensure all validations are applied!**

## Question 2- Lift System Automation

FAST university has hired you to make an operating system for their lifts. Below are specifications of the lifts OS.

The lift will be place in a building with 6 floors including a basement. From basement to top floors are labeled as -1,0,1,2,3,4 respectively. In the morning, the lift goes operation from the basement. The lift can go up and down. If maintained is required, the lift can be halted. If the lift is halted, the lift not usable during that period. Once unhalted, the lift can be used again.

With the help of above information, make a lift operating system which will have following functions

**int lift\_operating \_system(int requested\_floor, int current\_floor, char lift\_status)**

// this function validates the lift request and then performs the lift up or lift down function. Once the function is performed, the lift operating system returns the new current floor after lift goes up or down.

**int liftdown(int current\_floor,int requested\_floor )**

// will be called by lift\_operating\_system if list should move up

**int liftup(int current\_floor,int requested\_floor )**

// will be called by lift\_operating\_system if list should move dowb

**char halt\_lift(char status)**

//halts the lift, no up and down operation can be performed. Stored H for halting

**// char un\_half\_lift(char status)**

Unhalts the lift. Store W which represents that the lift is working.

**Main function:**

The main maintains the following is in an endless loop and asks the user to select on of the following operations.

1. go to a floor
2. Halt the lift
3. go to a floor
4. Unhalt the lift
5. go to a floor
6. Exit

Please note that all the validations are performed in **lift\_operating \_system**. The main is only the runner of the system. **Please ensure all validations are applied!**

### Question 3 - Matrix Multiplication

Write a C++ program that multiplies two matrix using functions. The program should include the following functions and specifications:

#### Specifications

- For matrix multiplication to be valid, the number of columns of the first matrix must equal the number of rows of the second matrix.
- Ensure that the program handles mismatched matrix sizes by displaying an appropriate error message.
- The matrices should be input and output in a readable format.
- Maximum size of matrix could be 10 x 10. Add **#define MAX 10** in your program.

#### Matrix and Input

- Define two, 2D array to ask user for the size of both matrixes (m(rows) x n(columns))
- Input all elements of the matrix

#### Matrix Multiplication Function:

- A function that multiplies two matrices. The function should take two matrices as parameters along with their sizes and return the resulting matrix of the multiplication.

```
void multiplyMatrices(int matrixA[MAX][MAX], int matrixB[MAX][MAX], int rowA, int colA, int rowB, int colB);
```

##### Parameters:

- matrixA[MAX][MAX]: The first matrix (A).
- matrixB[MAX][MAX]: The second matrix (B).
- rowA: The number of rows in matrix A.
- colA: The number of columns in matrix A.
- rowB: The number of rows in matrix B.
- colB: The number of columns in matrix B.

#### Display Function:

- A function to display the elements of a matrix. It should take a matrix and its size as parameters.

```
void displayMatrix(int matrix[MAX][MAX], int row, int col);
```

##### Parameters:

- matrix[MAX][MAX]: The matrix to be displayed.
- row: The number of rows in the matrix.
- col: The number of columns in the matrix.

#### Main Function:

- The main function should handle user input, call the above functions, and display the results.

### **Example Output**

Enter the number of rows and columns for the first matrix: 2 3

Enter the elements of the first matrix:

1 2 3

4 5 6

Enter the number of rows and columns for the second matrix: 3 2

Enter the elements of the second matrix:

7 8

9 10

11 12

Matrix A:

1 2 3

4 5 6

Matrix B:

7 8

9 10

11 12

Multiplication Result:

58 64

139 154