SET A

United International University

Department of Computer Science and Engineering

Final Examination Fall 2024

Course Code: **CSE 1112** Course Title: **Structured Programming Language Laboratory**
Date: Feb 11, 2025 (Tuesday) Time: 09:00 AM – 10:00 AM (1 hour) Full marks: 25

Name: Student ID:

Write down C programs for the following problems in Code Blocks (or any C compiler you prefer), and present the code to your instructor after the time is up. You can make rough calculations in this paper. **Any examinee found adopting unfair means will be expelled from the trimester/program as per UIU disciplinary rules.**

Problem 1 (Marks: 13)

Write a C program which **takes a string as user input**, **removes all the duplicate characters** present within the string and finally, **checks whether the string is a palindrome or not**. The individual tasks should be done within specific functions, which are mentioned in the following.

- **void removeAllDuplicates(char str[])**
  The function should receive a string as a parameter, and remove all duplicate characters present within said string.

- **int getSize(char str[])**
  The function should receive a string as parameter, and return the size of the string.

- **void checkPalindrome(char *str)**
  The function should receive a string as parameter, and check whether the string is a palindrome or not.

Note that the use of any predefined library functions is not allowed for solving the problem.

Sample Input	**Sample Output**
a	a palindrome
h	h palindrome
ssssss	s palindrome

ava	av not palindrome
madam	mad not palindrome
position	positn not palindrome

SET A

Problem 2 (Marks: 12)

Student Performance Management System

A university wants to develop a **Student Performance Management System** to keep track of students' academic performance. You need to implement a C program using structures to store and process student records. Each **student** has the following attributes: **name, id, marks, average.**

Your task is to implement the following functionalities:

1. **Add a New Student**: Input student details, including name, ID, and **marks for 5 subjects**. Compute and store their **average marks**.
2. **Display All Students**: Show the list of students along with their details and average marks.
3. **Find Top Performer**: Identify and display the student with the highest average marks.
4. **Find Failing Students**: Display students who have at least one subject where marks are below **40**. Also show the count on how many subjects he or she has failed.
5. **Exit**: Terminate the program.

Use an **array of structures** to store up to 100 students. Implement a **menu-driven approach** using if-else/switch case.

Sample Input/Output:

```
Student Performance Management System
1. Add a new student
2. Display all students
3. Find top performer
4. Find failing students
5. Exit
```

```
Enter your choice: 1
Enter student name: Alice
Enter student ID: 101
Enter marks for 5 subjects: 85 90 78 88 92
Student added successfully.

Enter your choice: 1
Enter student name: Bob
Enter student ID: 102
Enter marks for 5 subjects: 50 60 30 45 80
Student added successfully.

Enter your choice: 2
Student Records:
ID: 101, Name: Alice, Marks: [85, 90, 78, 88, 92], Average: 86.6
ID: 102, Name: Bob, Marks: [50, 60, 30, 45, 80], Average: 53.0

Enter your choice: 3
Top Performer: Alice, Average Marks: 86.6

Enter your choice: 4
Failing Students:
ID: 102, Name: Bob, Failed Subjects: 1

Enter your choice: 5
Exiting the program.
```

SET B

United International University

Department of Computer Science and Engineering

Final Examination Fall 2024

Course Code: **CSE 1112** Course Title: **Structured Programming Language Laboratory**
Date: Feb 11, 2025 (Tuesday) Time: 09:00 AM – 10:00 AM (1 hour) Full marks: 25

Name: Student ID:

Write down C programs for the following problems in Code Blocks (or any C compiler you prefer), and present the code to your instructor after the time is up. You can make rough calculations in this paper. **Any examinee found adopting unfair means will be expelled from the trimester/program as per UIU disciplinary rules.**

Problem 1 (Marks:13)

Implement a C program that p**rocesses a name and age** by checking if the name is a unique one, if yes, then modifying it under certain conditions, **and** checking if the age is a lucky number. You need to implement three functions to solve the problem:

1. **int hasUniqueCharacters(char *str)**: This function takes a string and checks whether all characters in the string are unique using only pointers (no array indexing allowed). It returns 1 if all characters are unique and 0 otherwise.
2. **void modifyString(char str[])**: If the string is contains all the unique character then this function removes all vowels and replaces them with the next consonant in the alphabet (e.g., 'a' → 'b', 'e' → 'f', etc.). This function **should not** use pointers.
3. **int isLuckyNumber(int num)**: This function takes an integer and returns 1 if the number is a **lucky number** (A number is a lucky number is it's a prime number and the age is odd). Otherwise, it returns 0.

Sample Input	Sample Output
Name: Alex Age: 13	Unique: YES Modified string: Blfx Lucky Number: YES

Name: Mili Age: 25	Unique: No Lucky Number: No

SET B

Problem 2 (Marks: 12)

Library Book Management System

You are tasked with developing a C program to manage book records in a library. Each book has details such as its **title, author name, publication year and availability status**. The program should provide users with the following operations through a menu-driven interface. For this task, you **must** implement the following functions:

- **Add a New Book (** void addBook(Book books[], int *count) **) :**
  This allows the user to enter the details of a new book including Title, Author Name, Publication Year and Availability Status and store them in the system.

- **Search for a Book by Title (** void searchBook(Book books[], int count, char title[]) **):**
  Search for a book by title. If found then display its Title, Author, Year and Status. Otherwise, display a "Book not available" message.

- **Update Book Availability Status (** void updateStatus(Book books[], int count, char title[]) **):**
  Update the availability status of a book by its title to "Available" or "Checked Out". If the book exists, the status is updated. Otherwise a "Book not found" message will be shown.

- **Display All Books (** void displayBooks(Book books[], int count) **):**
  Displays all books currently in the library. The list should **prioritize available books first** before displaying checked-out books.

- **Exit the Program:** This terminates the program.

You must first create a structure named Book to store the following details:

- Title (String)
- Author_Name (String)
- Publication_Year (Integer)
- Availability_Status (String: “Available” or “Checked Out”)

Instructions:

- Implement the menu using conditional statements (if-else or switch-case).
- Use an array of **Book** structures to store book records (maximum 100 books).
- Each functionality must be implemented as a separate function with the given function names and parameters.

Sample Input/Output

Library Management System:
1. Add a new book
2. Search for a book by Title
3. Update a book's availability status
4. Display all books
5. Exit

Enter your choice: 1
Enter book title: The Great Gatsby
Enter author name: F. Scott Fitzgerald
Enter publication year: 1925
Enter availability status (Available/Checked Out): Available
Output: Book added successfully.

Enter your choice: 2
Enter book title to search: The Great Gatsby
Output:
Title: The Great Gatsby, Author: F. Scott Fitzgerald, Year: 1925, Status: Available

Enter your choice: 3
Enter book title to update: The Great Gatsby
Enter new status (Available/Checked Out): Checked Out
Output: Book status updated successfully.

Enter your choice: 4
Output:
Title: The Great Gatsby, Author: F. Scott Fitzgerald, Year: 1925, Status: Checked Out

Enter your choice: 5
Output: Exiting the program.