The Open University of Sri Lanka
Faculty of Engineering Technology
Department of Electrical and Computer Engineering

Study Programme	:	Bachelor of Science Honours in Engineering
Name of the Examination	:	Final Examination
Course Code and Title	:	EEX3417 Software Development for Engineers
Academic Year	:	2024/25
Date	:	8th February 2026
Time	:	0930-1230h
Duration	:	3 hours

General instructions

1. Read all instructions carefully before answering the questions.
2. This question paper consists of **four(4)** questions on **4** pages.
3. Answer **all** questions.
4. The answer for each question should commence on a new page.
5. This is a Closed Book Test (CBT).

Question 01

(a) Use the pseudocode in Figure Q1.a to answer questions **(1)** and **(2)**.

```
START
DECLARE n, i, pos, neg
DECLARE arr[100]
DISPLAY "Enter number of elements"
INPUT n

SET i=0
SET pos=0
SET neg =0

WHILE i < n DO
   DISPLAY "Enter element", i + 1
   INPUT arr[i]

   IF arr[i] ≥ 0 THEN
      pos=pos + 1
   ELSE
      neg=neg + 1
```

```
    ENDIF

    i=i + 1
ENDWHILE

DISPLAY "Number of positive values =", pos
DISPLAY "Number of negative values =", neg

END
```

Figure Q1.a: pseudo code

(1) Draw the corresponding flowchart for the above pseudocode. **[10 marks]**

(2) Write a C program to implement the above pseudocode. The C program must use arrays, while loop, if–else statements, and input/output functions. Proper indentation and meaningful variable usage are expected.

[20 marks]

(b) The C program shown in Figure Q1.b is designed to input **integer** values into an **array**, calculate the **average** of those values, and display the **array elements**. However, some parts of the program are missing. Complete the program by filling in blanks (1) to (10) with the correct C keywords, identifiers, or symbols, writing only the blank number and the corresponding answer on your answer sheet.

[10 marks]

```
#_____(1)____ <stdio.h>
float readValues(int arr[], __(2)__ __(3)__)
{
   int total = 0;
   for (int __(4)__ = 0; i < size; i++)
   {
      printf("Enter value %__(5)__ : ", i + 1);
      scanf("%d", &arr[i]);
      total = total + __(6)__[i];
   }
   return (float)total / size;
}

void __(7)____(int arr[], int size)
{
   printf("Array elements are:\n");
   for (int i = 0; i < size; i++)
   {
      printf("%d ", arr[i]);
   }
   printf("\n");
```

```
}

int main()
{
   int size;
   float avg;
   ___(8)___("Enter array size: ");
   scanf("__(9)__", &size);

   int data[size];

   avg = readValues(data, size);
   showArray(data, size);

   printf("Average = %.2f", __(10)__ );
   return 0;
}
```

Figure Q1.b: C program

Question 02

Read the following description and answer questions (a) and (b).

A **School Library System** is designed using OOP to model library activities. It includes a base **Person** class inherited by **Student** and **Teacher** classes. **Students** can borrow **books**, while **teachers** add books to the **library**. The **Book** class stores book details, and the **Library** class manages books and teachers.
The system has the following requirements:

- *The **Person** class represents any person in the system. It has the attributes **name** and **age**, and a method **get_info()**.*
- *The **Student** class inherits from **Person**. It has the attribute **student_id** and a method **borrow_book(book)**.*
- *The **Teacher** class inherits from **Person**. It has the attribute **employee_id** and a method **add_book(book, library)**.*
- *The **Book** class contains the attributes **book_id**, **title**, **author**, and **year_published**.*
- *The **Library** class maintains a collection of **books** and a list of **teachers**. It includes the methods **add_teacher(teacher)** and **show_books()**.*

Actors	**Teacher:** Adds books, manages borrowing records. **Student:** Borrows books, views the list of available books.
Use Cases	**Add Book:** Teacher adds books to the system. **Manage Borrowing Records:** Teacher tracks borrowed books (includes **Verify Student**). **Verify Student:** Ensures valid student membership (included in **Borrow Book** and **Manage Borrowing Records**). **Borrow Book:** Student borrows a book (includes **Verify**

	Student, extends **Apply Fine**). **Apply Fine:** Applied if a student has overdue books (extends **Borrow Book**). **List Books:** Both Teacher and Student view available books.

(a) **Draw a UML Class Diagram** for the above system by clearly showing: All classes with their **attributes and methods, Inheritance relationships** between classes, **Composition** relationship between Library and Book, **Aggregation** relationship between Library and Teacher, Appropriate associations between Student, Teacher, and Library. **[15 marks]**

(b) Draw a **Use Case Diagram** for a **School Library System**. Include the system name, boundary, actors, use cases, and relationships (include/extend). **[15 marks]**

Question 03

Figure Q3.1 shows the MySQL table structures for the Car Details database.

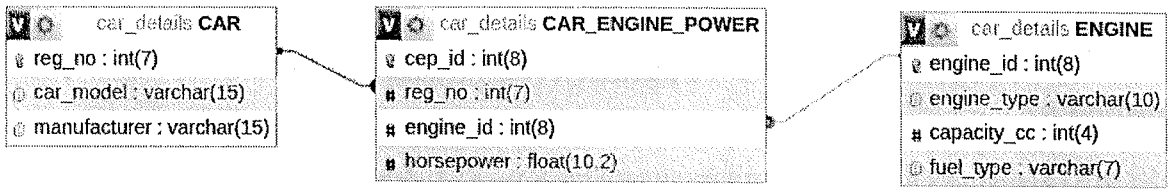


Figure Q3.1 MySQL tables for the Database

(a) Draw an **Entity Relationship Diagram (ERD)** for the tables shown in Figure Q3:1 with correct **entities, attributes, relationships, primary keys, and foreign keys.** **[10 marks]**

(b) Write the **database name, table names, field names with data types and lengths**, and clearly indicate the **primary keys and foreign keys** for each table. **[10 marks]**

Question 04

(a) Describe four types of software threats **[5 marks]**

(b) Explain what vulnerability is in your own words **[5 marks]**