# UNIVERSITY OF KELANIYA – SRI LANKA
# FACULTY OF SCIENCE

**Bachelor of Science (General) Degree (External)**
**First Year Second Semester Examination - 2020**

## Computer Science
## COSC 17543 (R) – Object Oriented Programming

**No of Questions: Five (04) No. of Pages: Three (03) Time: Two (02) Hours**

Answer **All** Questions.

---

1.

(a)

(i) List two differences between the Machine Language (ML) and High Level Programming Languages (HLL) (10 Marks)

(ii) List two drawbacks of Procedure Oriented Programming Languages (POPL). (10 Marks)

(iii) Provide a brief explanation of the following terms in Object-Oriented Programming (OOP): (20 Marks)

a. Class
b. Object
c. Attribute
d. Method

(b) Identify suitable classes, their attributes and methods of a Library Management System. (30 Marks)

(c) Identify the superclass, subclass(es), and draw the inheritance hierarchy for the following scenarios: (30 Marks)

(i) LivingOrganism, Plant, Animal, FloweringPlant, NonFloweringPlant, Insect, Mammal, and Bird classes.

(ii) ElectronicDevice, Computer, MobileDevice, Laptop, Desktop, Smartphone, and Tablet classes.

2.

(a) State whether each of the following statements is true or false. If your answer is false, provide a detailed explanation. (20 Marks)

(i) ***final*** is a keyword in Java.

(ii) Java identifier must begin with a letter, _ or $.

(iii) In Java, ***public***, ***private***, and ***protected*** are special data types.

(iv) ***int*** and ***long*** datatypes in java allows to store integer and floating-point values respectively.

(b) Define a Java class called Book according to the following specifications: (50 Marks)

(i) Book is represented by *ISBN*, *title*, *author*, *publisher*, *availableCopies*, and *totalCopies* attributes.

(ii) Add a default constructor to the class which assigns "*0000-0000*" to *ISBN*, "*Unknown*" to *title*, "*Unknown*" to *author*, "*Unknown*" to *publisher*, and *0* to both *availableCopies* and *totalCopies*.

(iii) Add six accessor methods to access the *ISBN*, *title*, *author*, *publisher*, *availableCopies*, and *totalCopies*.

(iv) Add six setter methods to set the *ISBN*, *title*, *author*, *publisher*, *availableCopies*, and *totalCopies*.

(v) Add two additional methods called borrowBook and returnBook to decrement the availableCopies when a book is borrowed and increment the availableCopies when a book is returned.

(c) Write a code fragment to declare and create two Book objects from the class defined in part (b) and name them *book1* and *book2*. Set the *ISBN* to "*1234-5678*" and "*9876-5432*", respectively. Set the *title* for both books to "*Java Programming*" and "*Data Structures*". Set the *availableCopies* to *5* and *2*, and the *totalCopies* to *5* and *2*, respectively. (10 Marks)

(d) Write a code fragment to declare and create another Book object from the class defined in part (b) and name it *book3*. Then, assign the *book2* object in part (c) to *book3*. (10 Marks)

(e) Draw the state-of-memory diagram for the *book1*, *book2*, and *book3* objects. (10 Marks)

3.

(a) Write a Java conditional statement that implements the table below, where *age* is an integer and *category* is a string. Both variables have been declared and *age* has been initialized to a value greater than or equal to 0: (25 Marks)

| If *age* is: | Set *Category* to |
|---|---|
| 0 – 2 | Toddler |
| 3 – 12 | Child |
| 13 – 19 | Teenager |
| 20 – 64 | Adult |
| 65 or greater | Senior |

(b) Write a Java switch statement for the table above that is identical to the statement in the previous problem. (15 Marks)

(c) Write a Java program that reads a student's test score (out of 100) and the number of assignments completed. Then, it calculates the final grade for the student. If the student has completed more than 8 assignments, they receive an additional bonus of 5% on their test score. However, the final grade cannot exceed 100. (40 Marks)

(d) Evaluate the following boolean expressions. For each of the expressions, assume a = 5, b = 10, and c = 15. Indicate whether each expression is always true, always false, or depends on the values of a, b, or c. (20 Marks)

(i) a + b > c && b - a < c

(ii) a * b >= c || c / b < a

(iii) !(a + c == b * 2) && !(b * c < a + b)

(iv) (a * 2 == b) || (b + c > 20 && a - b > 0)

4.

(a) What will be the value of sum after the following loops are executed? (40 Marks)

(i)
```
int count = 0, sum = 0;
while(count < 15 ) {
        sum += count;
        count++;
}
```

(ii)
```
int count = 0, sum = 0;
while(count < 30) {
        sum += 3*count;
        count += 2;
}
```

(iii)
```
sum = 0, j = 0;
do {
        j++;
        for (int i =4; i > j; i--)
                sum = sum + (i+j);
} while (j < 8);
```

(iv)
```
sum 0;
for (int i== 0; i <= 4; i++)
        for (int j 0; j<= 6; j++)
                sum += i;
```

(b) Declare a one-dimensional array of integers with size 30 to store the number of sales made each day by a salesperson over a 30-day period. Using this data structure, write a code fragment to implement the following tasks: (60 Marks)

(i) Display the number of sales made during the first week (days 1 to 7).

(ii) Calculate the total number of sales made over the entire 30-day period.

(iii) Find the day with the highest and lowest number of sales.

(iv) Retrieve the number of sales for any given day. The day is specified by an input value ranging from 1 to 30. Reject invalid input values (e.g., a number less than 1 or greater than 30).

(v) Calculate the average sales per week and display which week (out of the four weeks) had the highest and lowest average sales.

End of the Paper