CHUKA  **UNIVERSITY**

UNIVERSITY EXAMINATIONS

EXAMINATION FOR THE AWARD OF DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

ACSC 332: DESKTOP APPLICATION DEVELOPMENT

STREAMS: BSC (APPLIED COMP SCI) Y3S1 **TIME: 2 HOURS**

DAY/DATE: MONDAY 16/12/2024 **8.30 A.M – 10.30 A.M.**

INSTRUCTIONS TO CANDIDATES

- Answer Question **ONE** and any other **TWO** questions.
- Diagrams should be used whenever they are relevant to support an answer.
- There will be **No** use of mobile phones or any other unauthorized materials
- Write your answers legibly and use your time wisely
- Electronic, non-programmable calculators may be used

QUESTION ONE (COMPULSORY): 30 MARKS [ATTEMPT ALL QUESTIONS IN THIS SECTION

a. Software developers may use up to three layers for separation of concerns during the development process; Discuss why this concept is important. [6 marks]
b. With use of illustrations, describes the concept of polymorphism [4 marks]
c. Discuss the importance of JDBC and its application in java application [4 marks]
d. Discuss java collections and their application in building java applications [4 marks]
e. Explain the role that Java Foundation classes play in development of large softwares [4 marks]

f. Programmers developing desktop applications would prefer to use swing components over java AWT components. Describe advantages that each offer to a java developer. [4 marks]
g. Briefly discuss .NET framework and its important components in building desktop applications. [4 marks]

SECTION B: CHOOSE ANY TWO QUESTIONS FROM THIS SECTION

QUESTION TWO [20 MARKS]

a. The Java code below doesn't compile, explain what the programmer is missing to complete and compile the program [8 marks]

```
publicclassApp {
voidcalcBill() {
// Line n1
newInvoice().print();
    }
}
```

b. Discuss output of the software code below [12 marks]

```
importjava.util.Arrays;

publicclassApp {
publicstaticvoidmain(String[] args) {
String[] fruits = {
"banana",
"apple",
"pears",
"grapes"
        };
Arrays.sort(fruits, (a, b) ->a.compareTo(b));
for (String s: fruits) {
System.out.print(""+ s);
        }
    }
}
```

QUESTION THREE [20 MARKS]

a. With use of relevant code examples, demonstrate how each of the following concepts are implemented in java.

I.	Inheritance	[5 marks]
II.	Polymorphism	[5 marks]
III.	Encapsulation	[5 marks]
IV.	Abstraction	[5 marks]

QUESTION FOUR [20 MARKS]

A palindromic word is one that reads the same backwards as forwards. Hence the words hello and peel are not palindromes, but the words peep, deed, and aibophobia (fear of palindromes) are palindromes.

a) Define a class called Palindrome. **[2 marks]**

b) In your Palindrome class, create a method called reverse () which takes a string argument. Your method should return the reverse of the argument as a string. **[8 marks]**

c) Create a second method in Palindrome called is Palindrome() which takes a string argument. This method should return Trueif the argument is a palindrome and False otherwise.
[**10 marks]**

QUESTION 5 [20 MARKS]

1. Consider the following code in Object Oriented Programming which defines the start of a class to represent bank accounts:

```
Public class BankAccount{

Intinterest_rate = 0.3
Public BankAccount(String name, int number, double balance):
this.name = name;
this.number = number;
this.balance = balance;

return 0;
```

a) Add instance methods called deposit() and withdraw() which increase and decrease the balance of the account. **[4marks]**

b) Make sure the withdraw() method doesn't allow the account to go into overdraft. Add a third method called add interest () which adds interest to the balance (the interest should be the interest rate multiplied by the current balance). **[6 marks]**

c) Create a subclass of Bank Account called Student Account. Every Student Account should have an overdraft limit of Kshs 1000. **[4 marks]**

d) Write a constructor for the new class. Override the withdraw () method to make sure that students can withdraw money up to their overdraft limits. **[6 marks]**
