

**NANYANG TECHNOLOGICAL UNIVERSITY****SEMESTER 1 EXAMINATION 2021-2022****CE2002/CZ2002 – OBJECT-ORIENTED DESIGN & PROGRAMMING**

Nov/Dec 2021

Time Allowed: 2 hours

**INSTRUCTIONS**

1. This paper contains 4 questions and comprises 8 pages.
  2. Answer **ALL** questions.
  3. This is a closed-book examination.
  4. All questions carry equal marks.
- 

1. For each of the following multiple choice questions, choose the correct answer and provide brief justification.

- (a) Assume there is the following Square class with the attribute width and its `getWidth()` and `setWidth()` methods to read and change width. What is the output for the code below?

```
Square s1, s2;
s1 = new Square();
s2 = s1;
s2.setWidth(6);
s1.setWidth(99);
System.out.println(s2.getWidth());
```

- (1) 6  
 (2) 99  
 (3) 105  
 (4) none of the above

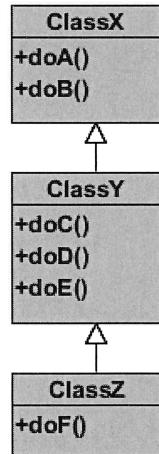
(4 marks)

Note: Question No. 1 continues on Page 2

- (b) In the code below, which line/s will have an error?
- ```
(i) public abstract class AnAbstract{
(ii)         private void doA(){System.out.print(); }
(iii)        private abstract void doB(int x); }
```
- (1) line (i)
  - (2) line (ii)
  - (3) line (iii)
  - (4) line (i) and line (ii)
  - (5) line (ii) and line (iii)
  - (6) line (i) and line (iii)
  - (7) all the lines

(4 marks)

- (c) Given the class hierarchy in **Figure Q1(c)**, which line will have a compilation error?

**Figure Q1(c)**

```

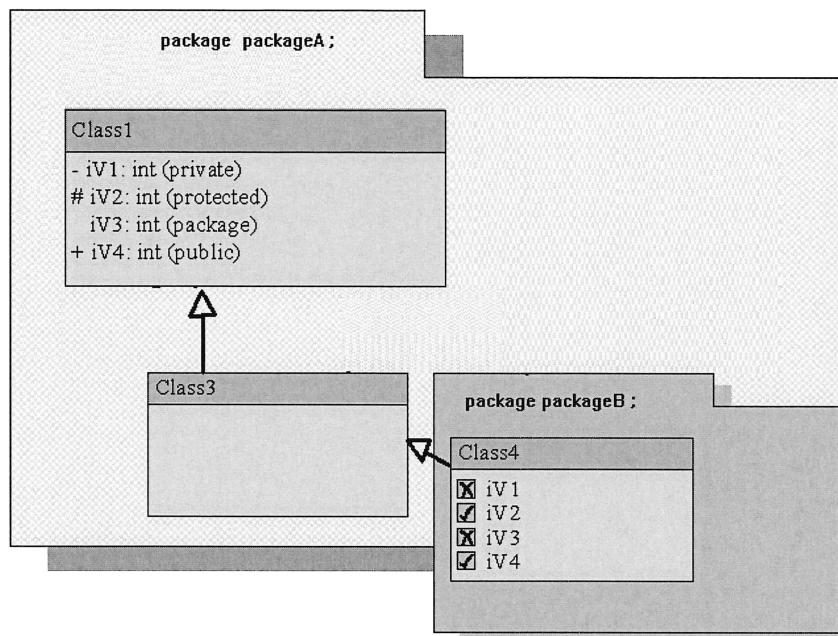
ClassX aa = new ClassY();
ClassZ bb = (ClassZ) aa;
ClassY cc = bb;
bb.doA();                      #line (i)
aa.doC();                      #line (ii)
cc.doB();                      #line (iii)
ClassY cc = bb;                #line (iv)
  
```

- (1) line (i)
- (2) line (ii)
- (3) line (iii)
- (4) line (iv)
- (5) none of the lines

(5 marks)

Note: Question No. 1 continues on Page 3

- (d) Looking at Class4 in **Figure Q1(d)**, which accessibility is wrong?



**Figure Q1(d)**

- (1) iV2
- (2) iV2 and iV3
- (3) iV3 and iV4
- (4) iV4
- (5) none

(4 marks)

- (e) The `talk(String x, int y)` method in `Student` class is showing \_\_\_\_\_.

```

public class Person{
    public void talk() {.....}
    public void talk(int x, String y) {.....}
}

public class Student extends Person{
    public void talk(String x, int y) {.....}
}

```

- (1) method overriding
- (2) method overloading
- (3) neither method overriding nor method overloading

(4 marks)

Note: Question No. 1 continues on Page 4

- (f) Given the Person and Student classes in **Question 1(e)**, the code below shows \_\_\_\_\_.
- ```
Person p = new Student();
p.talk(20,"hello");
```
- (1) downcasting, compile error  
 (2) upcasting, compile error  
 (3) upcasting, runtime OK  
 (4) downcasting, runtime OK
- (4 marks)
2. You are required to write a Java application program to calculate and show a student's overall score for a course he/she took. A course has several components such as an assignment, a research report, a final exam, etc. Each course component takes a certain percentage when calculating the overall score of a student for the course. A student can be an undergraduate student or a graduate student. A course taken by an undergraduate student has only two components, including an assignment and a final exam. A course taken by a graduate student has an additional component of a research report.
- (a) Write the code for the CourseComponent class that has the instance variables of componentName, percentage and score, a constructor, and the methods of getPercentage and getScore.
- componentName: the name of the course component.
  - Percentage: the percentage that the course component takes when calculating the overall score obtained by a student for the course.
  - Score: the score of the course component that a student obtains.
  - the constructor initializes the values of all instance variables.
  - getPercentage: accessor method to get the percentage value.
  - getScore: accessor method to get the score value.
- (5 marks)
- (b) Write the code for the **abstract** class Student that has the instance variable of name, a constructor, and the method of calOverall.
- name: the name of the student.
  - the constructor initializes the name of the student.
  - calOverall: an abstract method.
- (5 marks)

Note: Question No. 2 continues on Page 5

- (c) Two subclasses, Undergraduate and Graduate, are derived from the Student class. The Undergraduate class has two course components of assignment and finalExam, and the Graduate class has three course components of assignment, researchReport and finalExam. Both classes implement the method of calOverall that calculates the overall score of a course obtained by a student, which is the weighted average score of all the course components. Write a code for the two subclasses. (10 marks)
- (d) Write an application class StudentApp to have the printOverall method which prints out the average score of a course taken by a student. Also write the main function in the application class to demonstrate **dynamic binding** of the printOverall method for an undergraduate student and a graduate student. [Hint: having a single printOverall method in the application class, regardless of which Student subclass passed as argument of the printOverall method]. (5 marks)
3. (a) Study the following description of an automated teller machine (ATM):
- An ATM is an electronic banking outlet that allows customers to complete basic transactions without the aid of a branch representative or teller. Anyone with an ATM card can access cash at most ATMs. ATM Transaction means any actual or attempted use of an ATM, including to withdraw cash, deposit cash, transfer funds, or check account balances. Banks place ATMs inside and outside of their branches. Other ATMs are located in high-traffic areas such as shopping centers, grocery stores, convenience stores, airports, bus and railway stations, gas stations, casinos, restaurants.
- A bank rents the ATMs from Managed Service Providers but the branding is that of the bank.

Note: Question No. 3 continues on Page 6

ATMs require customers to use an ATM card, which contains Card Number to complete a transaction. Customers are authenticated by a Pin before any transaction can be made. A Customer inserts his/her ATM card and enters his/her pin, once his/her card details are validated by the bank then he/she can do the transaction. For a cash withdrawal transaction, first his/her bank account is checked to make sure there is sufficient balance in the account, then message goes to bank which reduces the bank account balance and sends a message to ATM to dispense cash. If Cash is dispensed successfully, it sends a message to the bank that transaction is completed successfully. The transaction information is logged with transaction ID, date, and transaction type after any transaction, including withdrawal transaction, cashDeposit transaction, fundTransfer transaction, or account balances inquiry.

The bank maintains customer's information including Name, Address, DOB, and account information. A customer might have multiple accounts with a bank. We assume that ATM transaction only access saving account.

You are tasked to identify the entity classes needed to build the application based on the description above.

Show your design with a Class Diagram. Your Class Diagram should show clearly the relationships between classes, enumeration, relevant attributes (at least TWO), logical multiplicities, meaningful role names, association names and constraint(s), if any. Showing class methods is optional.

(18 marks)

- (b) Explain your application of TWO of the SOLID design principles in your class diagram design in 3(a).  
(7 marks)
  
  
- 4. (a) Based on the ATM system described in Question 3, the simplified steps to withdraw money from an ATM using an ATM card are as follows.

Step 1: The customer inserts an ATM Card:

Step 2: The bank will verify the card. If the card is a valid card, the customer will be required to select a language from the language options appearing on the display screen and continue the transaction. If it is a stolen card, the card will be retained and the transaction terminates. If it is an invalid card, the card will be ejected and the transaction terminates.

Note: Question No. 4 continues on Page 7

- Step 3: After selecting language, the customer will be asked to enter 6-Digit ATM Pin. The bank will verify the Pin. If the Pin is correct, the customer will be asked to enter a withdrawal amount, or else, the user will be asked to enter the Pin again, if the customer entered the Pin wrong thrice, the card will be retained and the transaction terminates.
- Step 4: The ATM's remaining money will be checked, the transaction will be terminated if there is an insufficient money in the ATM to cover the withdrawal.
- Step 5: The bank will check customer's bank account to see if there is sufficient balance. If yes, withdrawal is successful, the balance in bank account will be updated, and the cash will be dispensed by the ATM, or else the transaction is unsuccessful due to insufficient balance, and the customer will be asked to enter a smaller amount, then the process will go back to step 4.
- Step 6: The card is ejected.

Draw the UML Sequence Diagram to show the flow of a user withdrawing money from ATM.

(15 marks)

- (b) Convert the following program in *java language* code to *C++ language* code by:
- Directly replacing with the *C++ language* syntax.
  - Write the code as *C++ classes*.
  - The *C++* program should work exactly same as the *Java* program.

```
abstract class iLivingthing{
    abstract public void grow();
}

class Animal extends iLivingthing {
    private int age;

    public Animal() {}

    public Animal(int age) {
        this.age = age;
    }

    public int getAge() {
        return age;
    }
}
```

Note: Question No. 4 continues on Page 8

```

public void grow() {
    System.out.println("Eat well to grow");
}

public void speak() {
    System.out.println("Sound");
}
}

class Cat extends Animal {
    private String name;

    public Cat(int age, String name) {
        super(age);
        this.name = name;
    }

    public void speak() {
        System.out.println("Meow");
    }
}

class Dog extends Animal {
}

public class Test {

    public static void main(String[] args) {
        Animal a1 = new Cat(2, "Snow");
        System.out.println(a1.getAge());
        a1.grow();
        a1.speak();
        Animal a2 = new Dog();
        System.out.println(a2.getAge());
        a2.grow();
        a2.speak();
    }
}

```

(10 marks)

END OF PAPER







**CE2002 OBJECT ORIENTED DESIGN & PROGRAMMING**  
**CZ2002 OBJECT ORIENTED DESIGN & PROGRAMMING**

Please read the following instructions carefully:

- 1. Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**
2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.
3. Please write your Matriculation Number on the front of the answer book.
4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.