Revision for Final Exam
SCSJ 2154

Class Relationship

Question 1 **[10 Marks]**

You are given the output of **Program B1** as in **Figure 1**.

```
Customer  id : CN79975174

List of Product:

1. YL010    Price per unit: RM 45.80
2. YL011    Price per unit: RM 78.80
```

Figure 1: Output of Program B1

a) Complete **Program B1** based on Questions (i) to (iv) below.
   i. Complete the definition of constructor for **Customer** class. (2 marks)
   ii. Write a code for customer buying product **p1.** (0.5 mark)
   iii. Write a code for customer buying product **p2**. (0.5 mark)
   iv. Write a code for displaying the information of customer and the products bought. (1 mark)

b) What is the relationship between **Customer** class and **Product** class? (1 mark)

c) Draw the UML class diagram for **Program B1** with correct attributes, operations and relationship. (5 marks)

```
1   //Program B1
2   class Product {
3      private String code;
4      private double price;
5
6      public Product(String code,double price) {
7         this.code = code;
8         this.price = price;
9      }
10     public String getProductCode() { return code; }
11     public double getPrice() { return price; }
12  }
13
14  class Customer
15  {
16     private String id;
17     private Product [] prod;
18     private int numOfProduct;
19
20
21     public Customer(String id) {
22        _____________________//
23        _____________________//(i)
24
```

```
25      }
26
27      public void buyProduct(Product p) {
28        prod[numOfProduct] = p;
29        numOfProduct++;
30      }
31
32      public void print() {
33        System.out.println("Customer id: " + id);
34        System.out.println("List of Product: ");
35        for (int i = 0; i < numOfProduct; i++) {
36          Product pd = (Product)prod[i];
37          System.out.print((i+1) + "." + pd.getProductCode());
38          System.out.println("\t Price per unit: RM " + pd.getPrice());
39        }
40      }
41
42  public class TestProduct {
43      public static void main(String arg[]) {
44        Product p1 = new Product ("YL010",45.80);
45        Product p2 = new Product ("YL011",78.80);
46
47        Customer cust1 = new Customer("CN79975174");
48        _____________________ //(ii)
49        _____________________//(iii)
50        _____________________//(iv)
51      }
52  }
```

Question 2 **[6 marks]**

Given the following Java programs, please answer questions (a) and (b).

```
1   //Program 6
2   public class Flight {
3      private int id;
4      private String origin;
5      private String destination;
6
7      public Flight(int id, String origin, String destination) {
8          System.out.println("Setting the Itinerary");
9          this.id = id;
10         this.origin = origin;
11         this.destination = destination;   }
12
13     public int getID() { return id; }
14     public String getOrigin() { return origin; }
15     public String getDestination() { return destination; }
16  }
```

Figure 6a: `Flight` class

```
1   //Program 7
2   public class FlightItinerary {
3      private Flight flight1;
4      public FlightItinerary() {
5          flight1 = new Flight(370,"KL","Beijing");   }
```

```
6
7       public void getItinerary() {
8           System.out.println("Flight ID:" + flight1.getID());
9           System.out.println("Flight Origin:" +
10                             flight1.getOrigin());
11          System.out.println("Flight Destination:" +
12                             flight1.getDestination());   }
13  }
```

Figure 6b: `FlightItinerary` class

```
1   //Program 8
2   public class TestBooking {
3       public static void main(String args[]) {
4           FlightItinerary fb = new FlightItinerary();
5           fb.getItinerary();  }
6   }
```

Figure 6c: `TestBooking` class

a) Based on the Program 6 to 8 given in Figure 6a, 6b, and 6c, draw the UML class diagram that shows the relationship between classes. [4 marks]

Answer:

b) What is the output of the programs? [2 marks]

Answer:

Question 3 **[10 marks]**

Given the following Java program, answer questions (a) to (d).

```
1   class Author {
2    private String name;
3    private Book[] myBook = new Book[10];
4    private int numOfBook;
5
6    public Author(String name){
7     this.name = name;}
8
9    public void write(Book bk){
10    _____________________________________________
11    _____________________________________________ }
12
13   public void print(){
14    System.out.println("AUTHOR NAME: " + name);
15    System.out.println("NUMBER OF BOOK(s) WRITTEN: " + numOfBook);
16    for(int i=0; i < numOfBook; i++) {
17       _____________________________________________
18       System.out.print((i+1) + ") " + s.getName());
```

```
19        System.out.printf("\tRM%.2f", s.getPrice());
20        System.out.println(); }}}
21  class Book {
22   private String name;
23   private double price;
24
25   public Book (String name, double price){
26    this.name = name;
27    this.price = price;}
28
29   public String getName(){
30    return name; }
31
32   public double getPrice(){
33    return price; }}
34
35  public class BookTest {
36   public static void main (String [] args) {
37    ____________________________________________
38    ____________________________________________
39    ____________________________________________
40    ____________________________________________ }}
41
```

a) Based on the program above, draw the UML class diagram that shows the relationship among classes. [3 marks]

b) Write codes for **`write()`** method (at line 10 to 11), to add objects from the **`Book`** class to a **`myBook`** array. The objects added to the array refer to the books written by an author. [2 marks]

c) Write a suitable statement at line 18 to complete the codes for **`print()`** method. [1 mark]

d) Write codes for **`main()`** method in **`BookTest`** class (at line 38 to 41) to create the object(s) of the **`Author`** and **`Book`** class. The codes should also be able to produce the output as shown in Figure 1. [4 marks]

```
AUTHOR NAME: Rosli Awang

NUMBER OF BOOK(s) WRITTEN: 1

1) Saat Berjaya RM45.70
```

Figure 1

Inheritance and Polymorphism

Question 1 **[10 marks]**

Given the following Program B2:

```
1   //Program B2
2   class ClassA{
3         public ClassA(){}
4         public void method1()
5         {     System.out.println("UTM"); }
6         public void method1(String a)
7         {     System.out.println("UTM" +a); }
8         public void method1(int a)
9         {     System.out.println("UTM" +a); }
10  }
11
12  class ClassB extends ClassA{
13        public ClassB(){}
14        public void method1()
15        {     System.out.println("FC UTM"); }
16        public void method1(String a)
17        {     System.out.println("FC UTM" +a); }
18        public void method2(String a, int b)
19        {     System.out.println("Studied at "+a+" in "+b); }
20  }
21
22  class ClassC extends ClassB{
23        public ClassC(){}
24        public void method1()
25        {     System.out.println("SE@FC UTM"); }
26        public void method1(int a)
27        {     System.out.println(" SE@FC UTM" +a); }
28  }
29
30  public class FinalPoly {
31  public static void main(String []args)
32  {
33  ____________________________________________
34  ____________________________________________
35
36  }
```

If the following statements are inserted at line 31 and 32, determine whether the program is correct or has an error during compilation. If the program is correct, state the output. If the program has an error, give the reason. Write your answer as in Table 1.

a)
```
ClassA ob = new ClassC();
ob.method1(2017);
```

b)
```
ClassA ob = new ClassC();
ob.method1("JB");
```

c)
```
ClassA ob = new ClassB();
ob.method2("FC UTM",2017);
```

```
d)  ClassC ob = new ClassB();
    ob.method1();

e)  ClassC ob = new ClassC();
    ob.method2("FSKSM UTM",1997);
```

Table 1

Statement No.	Correct / Error	Output/Reason
a)		
b)		
c)		
d)		
e)		

Question 2 **[10 marks]**

Figure 2 shows relationship of the classes in Program B3. Write the missing Java statements in Program B3 as guided in the comment parts in order to implement the class hierarchy as in Figure 2.

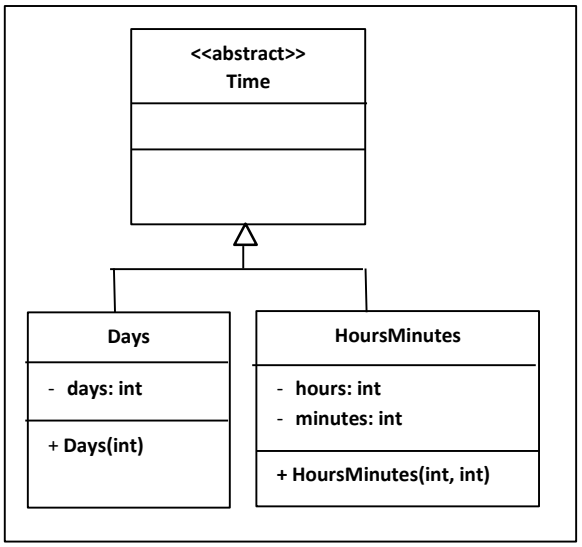

Figure 2

```
1   // Program B3
2   ___(i)____//Declaration of abstract class Time   (1M)
3   {
4   ___(ii)___//with an abstract method getMinutes() (1M)
5
6   ___(iii)__//Signature of class Days that inherits class Time (1M)
7   {
8           private int days;
9           ________(iv)________//Parameterized constructor
10          _________(v)________//of class Days (2M)
11
12          public int getMinutes() {
13              return days * 24 * 60;
14          }
15  }
16
17  ___(vi)___//Signature of class HoursMinutes that inherits class
18            //Time (1M)
19  {
20          private int hours;
21          private int minutes;
22
23          ___________(viii)_____//Parameterized constructor
24          ___________(ix)_______//of class HoursMinutes (2M)
25          ___________(x)________
26
27          public int getMinutes() {
28              return hours * 60 + minutes;
29          }
30  }
31
32  public class Demo {
33          public static void main(String args[]) {
34          //Create an object of class Time that refer to class Days
35          //named t1 with argument 2 (1M)
36              _____________(xi)___________
37          //Create an object of class Time that refer to class
38          //HoursMinutes named t2 with arguments 4 and 10 (1M)
39             __________(xii)_________________                  //
40             System.out.println(t1.getMinutes());
41             System.out.println(t2.getMinutes());
42          }
43      }
```

Question 3 **[10 marks]**

Given the UML class diagram in Figure 3, Program B4, and output in Figure 4, answer the following questions (a) to (d).

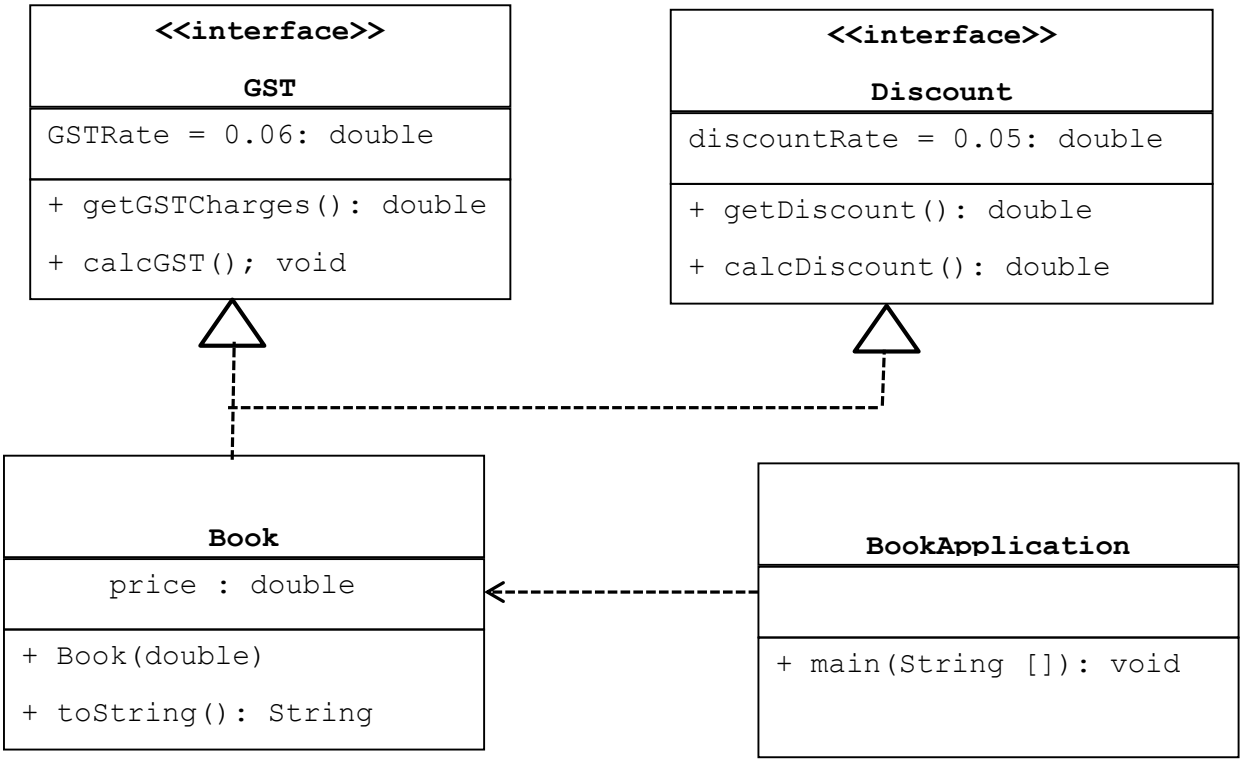

Figure 3: The UML class diagram

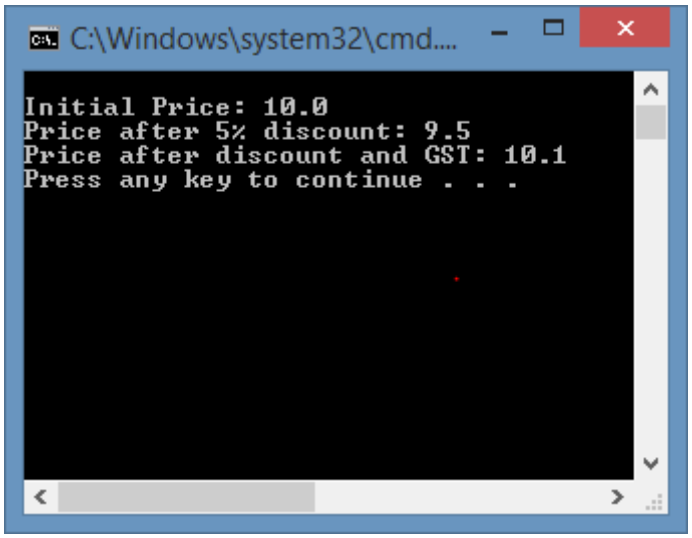

Figure 4: Output of Program B4

```
1   //Program B4
2
3       ________(a)__________{
4       ____________________________
5       ____________________________
6       ____________________________
7   }
8
9       ____________________ {
10      ____________________________
11      ____________________________
12      ____________________________
13  }
14
15  _________________(b)_____________________{
16  ____________________________________________
17  ____________________________________________}
18
19      public String toString(){
20        return "\nInitial Price: "+price+"\nPrice after 5% discount: " +
21        (price-calcDiscount())+ "\nPrice after discount and GST: "
22        +(price-calcDiscount()+ getGSTCharges());
23       }
24      public double getGSTCharges() {return price*RATE;}
25      public double calcGST() {return price+getGSTCharges();}
26      public double getDiscount() {return rate;}
27      public double calcDiscount() {return price*getDiscount();}
28  }
29
30  public class BookApplication {
31      public static void main(String[] args) {
32      ___________________(c)_____________________________
33  ___________________(d)_____________________________
34      }
35  }
```

a) Write Java code that defines **GST** (line 3-7) and **Discount** (line 11-15) interface classes. (5 marks)

b) Write Java code that defines class **Book** (line 18-22) that implements the interfaces defined in (a). (3 marks)

c) Write Java code to create a **Book** object with price is initialized with 10. (1 mark)

d) Display the price of book after discounts and tax levied **GST** by invoking **toString()** method. (1 mark)

Question 4 **[9 marks]**

a) The source codes in **Program B5** shows compile time error. Identify the statement that causing the error (give the line number). Explain the reason. (3 marks)

```
1   //Program B5
2   class Toyota
3   {
4       //Class Toyota Members
5   }
6
7   class Altis
8   {
9       //Class Altis Members
10  }
11
12  class Vios extends Toyota, Altis
13  {
14      //Class Vios Members
15  }
```

b) The source code in **Program B6** shows compile time error. Rewrite the program with the correction to the error. (3 Marks)

```
1   //Program B6
2   class Bird
3   {
4   public Bird(int i)
5       { System.out.println(1); }
6   }
7
8   class Eagle extends Bird
9   {
10  public Eagle()
11      { System.out.println(2); }
12  }
```

c) Give the output of **Program B7** below. (3 marks)

```
1   //Program B7
2   class Johor
3   {
4       { System.out.println(1);    }
5   }
6   class Skudai extends Johor
7   {
8       { System.out.println(2);    }
9   }
10  class Senai extends Skudai
11  {
12      { System.out.println(3);    }
13  }
14
15  public class MainClass
16  {
17    public static void main(String[] args)
18    { Senai c = new Senai();    }
19  }
```

Question 5 **[6 marks]**

Based on **Program B8**, answer the following questions.

```
1   //Program B8
2   class Concept1 {
3      public void tryMethod() {
4         System.out.println("Calling 1" );
5      }
6      public void tryMethod(int a) {
7         System.out.println("Calling 2, a: " + a);
8      }
9   }
10  class Concept2 extends Concept1 {
11     public void tryMethod(int b) {
12        System.out.println("Calling 3, b: " + b);
13     }
14     public void tryMethod(double a) {
15        System.out.println("Calling 4, a: " + a);
16     }
17  }
18  class TestConcepts      {
19     public static void main(String[] args) {
20        Concept2 ob = new Concept2();
21        double i = 7.0;
22        ob.tryMethod();
23        ob.tryMethod(15);
24        ob.tryMethod(i);
25     }
26  }
```

a) Identify which concept applied in tryMethod( ) function at line 11 of **Program B8** (overloading or overriding?). Justify your answer. (1.5 marks)

b) Which concept applied in tryMethod( ) at line 24 of **Program B8** (overloading or overriding?). Justify your answer. (1.5 marks)

c) Give the output of **Program B8.** (3 marks)

Question 6 **[8 marks]**

The following programs in **Card.java**, **Invitation.java** and **Festival.java** demonstrate the abstract concept. There are 4 errors in the programs. Rewrite the errors and its correct statements in a table format as exemplified in Table 2.

```
 1  // File: Card.java
 2
 3  public abstract superclass Card
 4  {
 5    private String receiver;
 6    public Card () {}
 7    public Card (String r)
 8    {   receiver = r; }
 9    public String getReceiver ()
10    {   return receiver; }
11    public abstract void wishes () {}
12  }
```

```
 1  // File: Invitation.java
 2
 3  class Invitation extends Card
 4  {
 5    private int age;
 6    public Invitation (String rec, int a) {
 7        superclass (rec);
 8        age = a;
 9    }
10
11    public void wishes () {
12        System.out.println("Special for: " +getReceiver());
13        System.out.println("Happy Birthday. Sweet" +age);
14    }
15  }
```

```
 1  // File: Festival.java
 2
 3  class Festival extends Card
 4  {
 5
 6    public Festival (String rec) {
 7        super (rec);
 8    }
 9
10    public abstract void wishes () {
11        System.out.println("Special for: " +getReceiver());
12        System.out.println("Greetings on your Special Day");
13    }
14  }
```

Table 2

File	Line of Error	Error Statement	Corrected Statement

Question 7 **[9 marks]**

You are given a UML class diagram in Figure 5. Complete the following program for class **Discount, UTM_Member** and **Student** by answering questions (a) to (d).

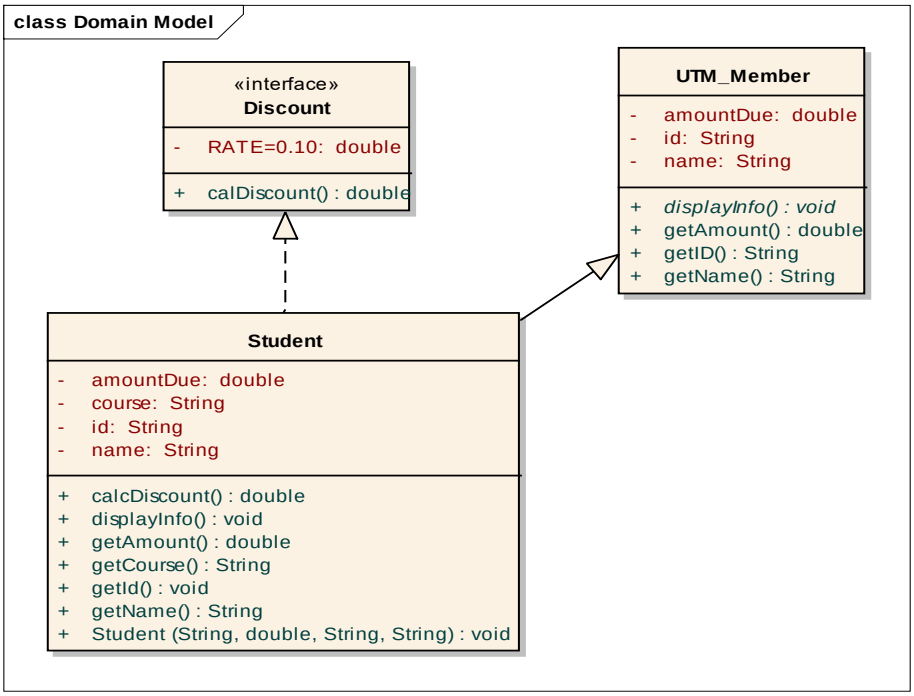

Figure 5

```
1   //(a)define interface Discount (1.5M)
2   {
3    _____________________
4   }
5   _____________________{//(b) Define class header for UTM_Member (1M)
6      private String name;
7      private String id;
8      private double amountDue;
9
10     public UTM_Member(String name, String id, double amountDue) {
11         this.name = name;
12         this.id = id;
13         this.amountDue = amountDue;
14     }
15
16     public String getName()
17     { return name; }
18
19     public String getId()
20     { return id;    }
21
22     public double getAmount()
23     { return amountDue; }
24
25     ________________//(c)define an abstract method displayInfo() (1M)
26  }
26
27  //(d)define class Student with attributes & constructor (3.5M)
28  { _____________________
29
30
31     public String getCourse() {
32         return course;    }
33
34     public double calcDiscount()
```

```
35      {  double discAmount = getAmount()*RATE;
36         return discAmount;      }
37
38      public void displayInfo()
39      {
40         System.out.println ("Student Name: "+ getName());
41         System.out.println ("Student ID : "+ getId());
42         System.out.println ("Course : "+ getCourse());
43         System.out.println ("Amount Due: RM"+ getAmount());
44         System.out.printf ("Total price after discount : RM %.2f \n",
45         (getAmount()- calcDiscount()));
46      }
47  }
```

e) What will happen if method **`calcDiscount()`** is not overridden in class **`Student`**? (2 marks)

Question 8 **[10 marks]**

Given the UML class diagram in Figure 9 that represents the Java programs: Program 14 to 16 (shown in Figure 10), please answer questions (a) to (c).

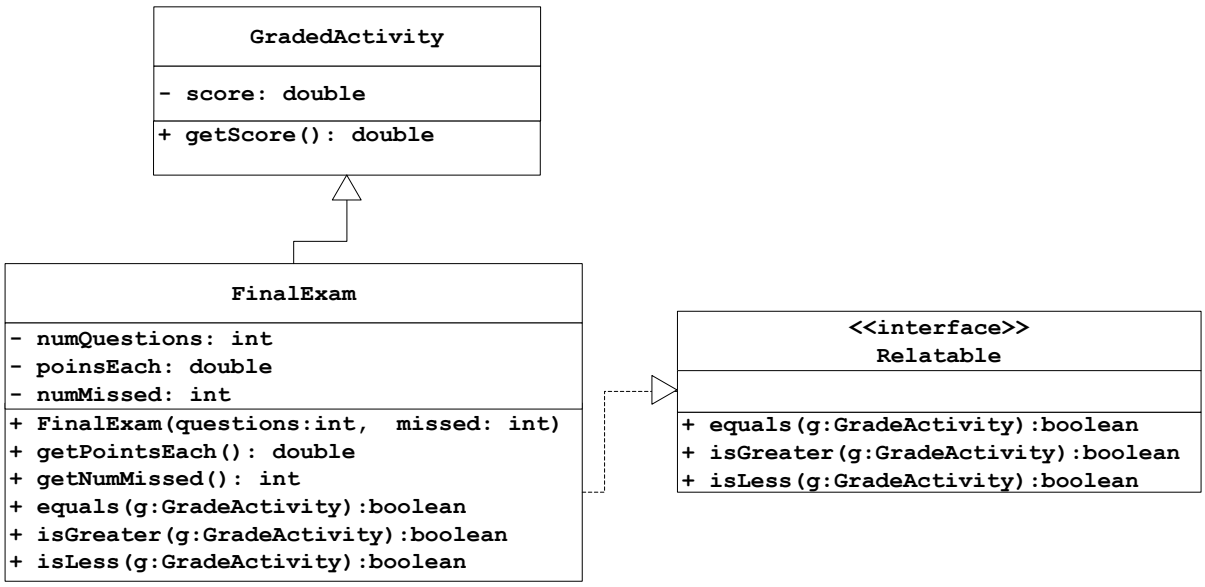

Figure 9: The UML class diagram

```
1   //Program 14
2   public class FinalExam        For question 4(b)        {
3      private int numQuestions;
4      private double pointsEach;
5      private int numMissed;
6
7      public FinalExam(int questions, int missed) {
8         double numericScore;
9         numQuestions = questions;
10        numMissed = missed;
11
12        pointsEach = 100.0 / questions;
13        numericScore = 100.0 - (missed * pointsEach);
```

```
14
15          setScore(numericScore);
16       }
17
18       public double getPointsEach() { return pointsEach; }
19       public int getNumMissed() { return numMissed; }
20
21       public boolean equals(GradedActivity g) {
22          boolean status;
23
24          if (this.getScore() == g.getScore())
25             status = true;
26          else
27             status = false;
28
29          return status;
30       }
31
32       public boolean isGreater(GradedActivity g) {
33          boolean status;
34
35          if (this.getScore() > g.getScore())
36             status = true;
37          else
38             status = false;
39
40          return status;
41       }
42
43       public boolean isLess(GradedActivity g) {
44          boolean status;
45
46          if (this.getScore() < g.getScore())
47             status = true;
48          else
49             status = false;
50
51          return status;
52       }
53    }
```

```
1     //Program 15
2     public class GradedActivity {
3        private double score;
4        public double getScore() { return score; }
5     }
```

```
1     //Program 16
2     public class InterfaceDemo {
3        public static void main(String[] args) {
4           FinalExam exam1 = new FinalExam3(100, 20);
5           FinalExam exam2 = new FinalExam3(100, 30);
6
7           System.out.println("Exam 1: " + exam1.getScore());
8           System.out.println("Exam 2: " + exam2.getScore());
9
10          _____ For question 4(c) _____ }
11    }
```

Figure 10: Java program

a) Write an interface definition named **`Relatable`**, based on the UML class diagram shown in Figure 9. [4 marks]

b) In Program 14, line 2, complete the header of **`FinalExam`** class declaration. **`FinalExam`** class is the subclass inherits from the **`GradedActivity`** superclass, and implements the **`Relatable`** interface. [1.5 marks]

c) Complete the **`InterfaceDemo`** class program (Program 16), by writing Java statements to display a messages based on condition stated in Table 3. ***Note:*** You are not allowed to use conditional operator to display the messages. You must use a suitable methods that you are already created in **`FinalExam`** class. [4.5 marks]

Table 3: Message to print

Condition	Message(s) to Print
`exam1 = exam2`	The exam scores are equal.
`exam1 > exam2`	The exam 1 score is the highest.
`exam1 < exam2`	The exam 1 score is the lowest.

Exception Handling

Question 1 **[10 marks]**

a) Answer question (i) to (v) as in Program B9 below with suitable codes so that it can throw the exception. (5 marks)

```
//Program B9
  public class FinalExamException {
    public static void main (String args[]) {
      int arr []={30,40};
      Scanner in= new Scanner (System.in);
    ____(i)____ {
    int b = in.nextInt();
    int x = arr[2]/ (b – arr[1]);
   }

   catch (_____(ii)_____ ex) {
   System.out.println("Exceed array size”);
   }
   catch (_____(iii)_____ex) {
   System.out.println("Denominator is zero”);
   }

   catch (_____(iv)_____ex) {
   System.out.println("Invalid data:” +e);
   }

   _____(v)_____ {
   int y = arr[1]/ arr[0];
   System.out.println("y = ” + y);
   }
  }
}
```

b) Given Program B10 below, answer the following question.

i. Explain why error will occur when Program B10 is compiled? [2 marks]

ii. Rewrite the program so that the program will compile and run properly. [3 marks]

```
//Program B10
class Test {
  public static void main(String[] args) {
    try {
      String s = "5.6";
      Integer.parseInt(s);

      int i = 0;
      int y = 2 / i;
    }
    catch (Exception ex) {
      System.out.println("NumberFormatException");
    }
    catch (RuntimeException ex) {
      System.out.println("RuntimeException");
    } } }
```

Question 2 **[8 marks]**

Answer question (i) to (v) as in **`Program B11`** below with suitable codes to produce output as in Figure 6.

```
//Program B11
public class TestException {
   public static void main (String args[]) {
      int array[]={20,20,40};
      int num1=15, num2=0;
      int result=10;
      __(i)_ {
         result = num1/num2;
         System.out.println("The result is" +result);
         for(int i =5;i >=0; i--) {
            System.out.println("The value of array is"
            +array[i]);
         }
      }
      ____(ii)____(______(iii)_________ ex) {
         System.out.println("Array is out of Bounds”);
      }
      ____(iv)____(_______(v)_________ ex) {
         System.out.println ("Can't divide by Zero");
      }
   }
}
```

```
Can't divide by Zero
```

Figure 6

Question 3 **[10 marks]**

Given the following Java program, please answer questions (a) and (b).

```
1  //Program 17
2  import java.io.*;
3  public class MyFinal
4  {
5     static void exceptionFinal() throw RuntimeException eE  {
6         System.out.println("Inside exceptionFinal");
7         throw RuntimeException();
8     }
9
10
11    public static void main(String args[])
12    {
13       double myDouble[] = new double[5];
14
15       try  {
16           exceptionFinal();
17           System.out.println("Access element sixth :" +
18           myDouble[6]);
19        }
```

```
20          catch (RuntimeException eE)  {
21              System.out.println("Exception thrown: 1");
22           }
23          catch (Exception eE)  {
24              System.out.println("Exception thrown: 2");
25           }
26          catch (ArrayIndexOutOfBoundsException eE)  {
27              System.out.println("Exception thrown: 3" );
28           }
29           finally  {
30              System.out.println("Exception end" );
31          }
32          System.out.println("Out of the block");
33      }
34  }
35
```

a) There are six (6) errors in the class **`MyFinal`** in Program 17. Identify them, by write the line number of code that contains the error and the suitable code to correct the error. State your answers in the Table 4. [6 marks]

Table 4

Line	Cause of Error	The Correct Code

b) What will be the output of the Program 10 after the errors in (a) have been corrected. [4 marks]

Answer: