



| **SUBJECT CODE** | **: SCSJ2154** |
| --- | --- |
| **SUBJECT NAME** | **: OBJECT ORIENTED PROGRAMMING** |
| **YEAR/COURSE** | **: 2 (SCSJ / SCSV / SCSB / SCSR)** |
| **TIME** | **: 1½ Hours** |
| **DATE** | **:** |
| **VENUE** | **:** |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# INSTRUCTIONS :

This test book consists of 2 parts:

Part A: 10 Multiple Choice Questions 10 marks

Part B: 4 Questions 40 marks

**ANSWER ALL QUESTIONS IN THE ANSWER BOOKLET.**

**(Please Write Your Lecture Name And Section In Your Answer Booklet)**

| Name |  |
| --- | --- |
| I/C No. |  |
| Year / Course |  |
| Section |  |
| Lecturer Name |  |

# This questions paper consists of ELEVEN ( 11 ) printed pages excluding this page.

**SECTION A: OBJECTIVE QUESTIONS (10 MARKS)**

*Part A consists of 10 objective questions. Choose the best answer, and write your answer in the answer booklet. Each question carries 1 mark.*

1. Object oriented inheritance models the

| A. | "is a kind of" relationship |
| --- | --- |
| B. | "has a" relationship |
| C. | "want to be" relationship |
| D. | “contains” of relationship |

1. What is the output of the following code?



| 1. 0 | 1. 1 |
| --- | --- |
| 1. 2 | 1. Nothing |

1. Which of the following declares an abstract method in an abstract Java class?

| A. | public abstract method(); |
| --- | --- |
| B. | public abstract void method(); |
| C. | public void abstract method(); |
| D. | public abstract void method() {} |

1. Which of the following statements is appropriate to be filled in the blank space in **Program A1** below?



| 1. super(t);   title=t; | 1. super(p);   price=p; |
| --- | --- |
| 1. super(t);   price=p; | 1. super(p);   title=t; |

1. Consider the following class definition:

**public class Student**

**{**

**protected int x;**

**public abstract double print();**

**public void setX(int a)**

**{ x = a; }**

**public class Student()**

**{ x = 0; }**

**}**

What is wrong with the class definition?

| A. | The keywords public and abstract cannot be used together. |
| --- | --- |
| B. | The method **print()** in class **Student** must have a body. |
| C. | Class **Student** must be defined abstract. |
| D. | Variable x cannot be declared as protected. |

1. Which of the following is a correct definition of interface class A?

| A. | interface A { void print() { }; } |
| --- | --- |
| B. | abstract interface A { print(); } |
| C. | abstract interface A { abstract void print() { };} |
| D. | interface A { void print();} |

1. Suppose A is an interface, B is a concrete class that implements A. Which of the following is correct?

| i. | A a = new A(); |
| --- | --- |
| ii. | A a = new B(); |
| iii. | B b = new A(); |
| iv. | B b = new B(); |

| 1. i and ii only. | 1. ii and iv only. |
| --- | --- |
| 1. i and iii only. | 1. all above. |

1. An instance of \_\_\_\_\_\_\_\_\_ describes system errors. If this type of error occurs, there is little you can do beyond notifying the user and trying to terminate the program gracefully.

| A. | RuntimeException |  |  | C. | Error |
| --- | --- | --- | --- | --- | --- |
| B. | Exception |  |  | D. | Throwable |

1. What will happen when **Program A2** is compiled and run?



| i. | An exception is raised due to Integer.parseInt(s); |
| --- | --- |
| ii. | An exception is raised due to 2/i; |
| iii. | The program has a compilation error. |
| iv. | The program compiles and runs without exceptions. |

| 1. i and ii only. | 1. ii only. |
| --- | --- |
| 1. iii only. | 1. iv only.. |

1. What is wrong with the following program?



| A. | You cannot have a try block without a catch block. |
| --- | --- |
| B. | You cannot have a try block without a finally block. |
| C. | A method call that does not declare exceptions cannot be placed inside a try block. |
| D. | Nothing is wrong. |

**SECTION B: STRUCTURED QUESTIONS (40 MARKS)**

*Part B consists of 4 structured questions. Answer all questions in the answer booklet. The marks for each part of the question is as indicated.*

**Question 1 [10 marks]**

Given the following Program B1;

| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27**  **28**  **29**  **30**  **31**  **32**  **33**  **34** | **//Program B1**  **class ClassA{**  **public ClassA(){}**  **public void method1()**  **{ System.out.println("UTM"); }**  **public void method1(String a)**  **{ System.out.println("UTM" +a); }**  **public void method1(int a)**  **{ System.out.println("UTM" +a); }**  **}**  **class ClassB extends ClassA{**  **public ClassB(){}**  **public void method1()**  **{ System.out.println("FC UTM"); }**  **public void method1(String a)**  **{ System.out.println("FC UTM" +a); }**  **public void method2(String a, int b)**  **{ System.out.println("Studied at "+a+" in "+b); }**  **}**  **class ClassC extends ClassB{**  **public ClassC(){}**  **public void method1()**  **{ System.out.println("SE@FC UTM"); }**  **public void method1(int a)**  **{ System.out.println(" SE@FC UTM" +a); }**  **}**  **public class TestFC {**  **public static void main(String []args)**  **{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **}** |
| --- | --- |

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 B1 shows relationship of the classes in Program B2. Write the missing Java statements in Program B2 as guided in the comment parts in order to implement the class hierarchy as in Figure B1.



**Figure B1**

| **1.**  **2.**  **3.**  **4.**  **5.**  **6.**  **7.**  **8.**  **9.**  **10.**  **11.**  **12.**  **13.**  **14.**  **15.**  **16.**  **17.**  **18.**  **19.**  **20.**  **21.**  **22.**  **23.**  **24.**  **25.**  **26.**  **27.**  **28.**  **29.**  **30.**  **31.**  **32.**  **33.**  **34.**  **35.**  **36.**  **37.**  **38.**  **39.**  **40.**  **41.**  **42.**  **43.**  **44.**  **45.** | **// Program B2**  **\_\_\_\_\_\_\_\_\_\_\_\_\_(i)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ // Declaration of abstract class Time**  **{ //** [1 marks]  **\_\_\_\_\_\_\_\_\_\_\_\_\_(ii)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ // with an abstract method**  **} // getMinutes()** [1 marks]    **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(iii)\_\_\_\_\_\_\_\_\_\_\_\_ // Signature of class Days that**  **{ // inherits class Time** [1 marks]  **private int days;**  **\_\_\_\_\_\_\_\_(iv)\_\_\_\_\_\_\_\_\_\_\_\_ // Parameterized constructor**  **\_\_\_\_\_\_\_\_\_(v)\_\_\_\_\_\_\_\_\_\_\_\_ // of class Days** [2 marks]    **public int getMinutes() {**  **return days \* 24 \* 60;**  **}**  **}**    **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(vi)\_\_\_\_\_\_\_\_\_\_\_ // Signature of class HoursMinutes that**  **{ // inherits class Time** [1 marks]  **private int hours;**  **private int minutes;**  **\_\_\_\_\_\_\_\_\_\_\_(viii)\_\_\_\_\_\_\_\_\_ // Parameterized constructor**  **\_\_\_\_\_\_\_\_\_\_\_(ix)\_\_\_\_\_\_\_\_\_\_\_ // of class HoursMinutes** [2 marks] **\_\_\_\_\_\_\_\_\_\_\_\_(x)\_\_\_\_\_\_\_\_\_\_\_**    **public int getMinutes() {**  **return hours \* 60 + minutes;**  **}**  **}**  **public class Demo {**  **public static void main(String args[]) {**  **// *Create an object of class Time that refer to class Days***  **// *named t1 with argument 2***  **\_\_\_\_\_\_\_\_\_\_\_\_\_(xi)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //** [1 marks]  **// *Create an object of class Time that refer to class* HoursMinutes**  **// *named t2 with arguments 4 and 10***  **\_\_\_\_\_\_\_\_\_\_(xii)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //** [1 marks]  **System.out.println(t1.getMinutes());**  **System.out.println(t2.getMinutes());**  **}**  **}** |
| --- | --- |

**Question 3 [10 marks]**

Given the UML class diagram in Figure B2, Program B3, and output in Figure B3, answer the following questions (a) to (c).









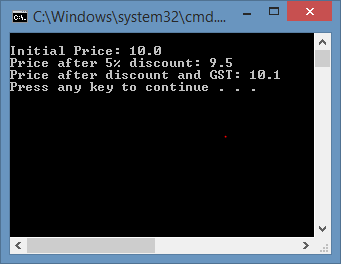








**Figure B2:** The UML class diagram



**Figure B3 : Output of Program B3**

| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27**  **28**  **29**  **30**  **31**  **32**  **33**  **34**  **35**  **36**  **37**  **38**  **39**  **40**  **41** | **//Program B3**  **\_\_\_\_\_\_\_(a)\_\_\_\_\_\_\_\_\_{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **}**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ {**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **}**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_(b)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_{**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_}**  **public String toString(){**  **return "\nInitial Price: "+price+"\nPrice after 5% discount: "+ (price-calcDiscount())+ "\nPrice after discount and GST: "+(price-calcDiscount()+getGSTCharges()); }**  **public double getGSTCharges() {return price\*RATE;}**  **public double calcGST() {return price+getGSTCharges();}**  **public double getDiscount() { return rate;}**  **public double calcDiscount() { return price\*getDiscount();}**  **}**    **public class BookApplication {**  **public static void main(String[] args) {**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(c)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(d)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **}**  **}** |
| --- | --- |

1. Write Java code that defines GST (line 3-7) and Discount (line 11-15) interface classes [5 marks]
2. Write Java code that defines class Book (line 18-22) that implements the interfaces defined in (a). [3 marks]
3. Write Java code to create a Book object with price is initialized with 10. [ 1 mark]
4. Display the price of book after discounts and tax levied GST by invoking toString() method. [1 mark]

**Question 4 [10 marks]**

1. Answer question (i) to (v) as in Program B4below with suitable codes so that it can throw the exception. [ 5 marks]

| **//Program B4**  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);  }  }  } |
| --- |

1. Given Program B5 below, answer the following question.
   * 1. Explain why error will occur when Program B5 is compiled? [ 2 marks]
     2. Rewrite the program so that the program will compile and run properly. [ 3 marks]

| **//Program B5**  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");     }   } } |
| --- |