



# EAST WEST UNIVERSITY

## Department of Computer Science and Engineering

### B.Sc. in Computer Science and Engineering Program

### Mid Term II Examination, Fall 2020 Semester

**Course:** CSE110-Object Oriented Programming, Section-1

**Instructor:** Tanni Mittra, Senior Lecturer, CSE Department

**Full Marks:** 30 (15 will be counted for final grading)

**Time:** 1 Hour and 20 Minutes

**Note:** There are 5 (five) questions, answer ALL of them. Course Outcome (CO), Cognitive Level and Mark of each question are mentioned at the right margin.

1. **Define** a class *Customer* that has two instance variables: *cusId* and *Name*. Include appropriate constructors. Also include setters and getters if required. Add *toString()* method that returns a string containing customer's information. [CO2, C3, Mark: 6]

**Define** another class *Teller* that has four instance variables: *Id*, *date*, *slotId* (each teller will serve 10 customers in a date) and customer information. Include appropriate constructors. Add *toString()* method that returns a string containing teller's information. Moreover, the *Teller* class must have an *addcustomer(Customer c)* method which is invoked when a customer wants to withdraw money. Please note that, when a customer is added *slotId* will be incremented by one. If *slotId* is 10 then no other customer will be added to the customer list.

**Write** a *Main* class that includes a *main()* method for testing the functionalities of these two classes: *Customer* and *Teller*. Create an *ArrayList* of 11 Customers and add the customers to the customer list of a *Teller* object. Remember that, for 11<sup>th</sup> customer print a message "slots are not available".

2. A hospital has mainly two types of patients: Outdoor patients are coming to visit a doctor and pays a visiting fee and Indoor patients are admitted into the hospital and pays fee in different categories such as bed charge, medicine charge, lab test charge etc. The hospital wants you to write a Java application that performs its bill calculations polymorphically considering the following specifications. [CO2, C3, Mark: 3]

Patient class is an abstract class and has two instance variables: *id* and *name*.

OutdoorPatient class has one instance variable of its own, which is visitingFee. IndoorPatient has three instance variables: bedFee, medicineFee and labTestFee. All classes include appropriate constructors and toString() method. Patient class also include an abstract method bill().

Draw the corresponding class diagram. Use appropriate access specifier symbol, data type and return type of methods.

3. As per the use case specified in Question 2, *bill()* and *toString()* methods return the followings: [CO2, C3, Mark: 10]

Class	<i>bill()</i> Method	<i>toString()</i> Method
<i>Patient</i>	abstract method	<i>id name</i>
<i>OutdoorPatient</i>	<i>visitingFee</i>	<i>id name</i> visiting fee: <i>visitingFee</i>
<i>IndoorPatient</i>	returns the sum of <i>bedFee</i> , <i>medicineFee</i> and <i>labTestFee</i>	<i>id name</i> bed fee: <i>bedFee</i> medicine fee: <i>medicineFee</i> lab test fee: <i>labTestFee</i>

**Define** *Patient* abstract class. Include constructor(s), toString() and an abstract method *bill()*.

**Define** *OutdoorPatient* class. Include constructor(s) and override bill() and toString() methods as per the specification shown in the table above.

**Define** *IndoorPatient* class. Include constructor(s) and override bill() and toString() methods as per the specification shown in the table above.

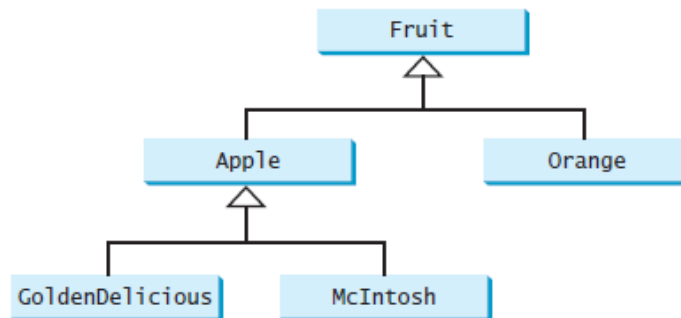
**Write** a *Main* class that has *main()* method. Within the *main()*, define an ArrayList of appropriate type. Instantiate two objects each from *IndoorPatient* and *OutdoorPatient* class and add those four objects into that array list. Invoke *bill()* and *toString()* methods polymorphically and print these values. At last, print the average bill of *IndoorPatient* type patients.

4. Consider the use case described in Question 2. [CO2, C3, Mark: 6]
- If a new class Extend *OutdoorPatient* and *IndoorPatient* class. What type of inheritance is used here for extending these classes? Is it possible to do it in JAVA? Justify your answer.
  - Now change *OutdoorPatient* as abstract class and write another class *ReportPatient* that will implement *OutdoorPatient* class. Do the necessary

actions to extend the class.

5. Consider the following inheritance hierarchy:

[CO2, C3,  
Mark: 5]



**Answer** the following questions with proper justification:

Assume that the following code is given:

```

Fruit fruit = new GoldenDelicious();
Orange orange = new Orange();
  
```

Answer the following questions with a brief justification:

- Is fruit instanceof GoldenDelicious?
- Suppose the method makeOrangeJuice is defined in the Orange class. Can orange invoke this method? Can fruit invoke this method?
- Is the statement GoldenDelicious p = new Apple() legal?
- If Fruit class contains a method eat(), then if we write the following code inside Orange class,  
`Super.eat();`  
 Is the following statement is correct to invoke the method of Fruit class.
- If we declare another class Malta which extends both Apple and Orange class. Is the above scenario is possible? Justify your answer.