**Object Oriented Programming in Java (Lab 4)**

**Problem:**

1. (The Person , Student , Employee , Faculty , and Staff classes) Design a class named Person and its two subclasses named Student and Employee . Make Faculty and Staff subclasses of Employee . A person has a name, address, phone number, and email address. A student has a class status (freshman, sophomore, junior, or senior). Define the status as a constant. An employee has an office, salary, and date hired. Define a class named MyDate that contains the fields year , month , and day . A faculty member has office hours and a rank. A staff member has a title. Override the toString method in each class to display the class name and the person’s name. Draw the UML diagram for the classes. Implement the classes. Write a test program that creates a Person, Student, Employee, Faculty and Staff and invokes their toString() methods.

2. Define the Triangle class with three sides. In a triangle, the sum of any two sides is greater than the other side. The Triangle class must adhere to this rule. Create the IllegalTriangleException class, and modify the constructor of the Triangle class to throw an IllegalTriangleException object if a triangle is created with sides that violate the rule, as follows:

/\*\* Construct a triangle with the specified sides \*/

public Triangle(double side1, double side2, double side3)

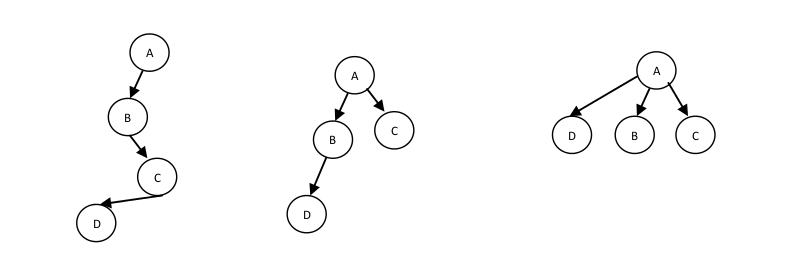
throws IllegalTriangleException {

// Implement it

}

3. Explore Exception propagation. Define a chain of calling few methods such that each one has a routine to handle some exception but not all. Observe how different exceptions thrown in the bottom most called module in the calling chain handled.

4. Explore Dynamic Binding. Define some classes say A, B, C etc. following different hierarchy as follows. And have a method having same name and arguments with different definition such that it is defined separately in only some of the classes



**Structure:**

1. Package PersonInheritance contains 6 classes: Person, Student, Employee, Staff, Faculty and

Mydate.

Default Package contains the Inheritance class which contains the main function for the program.

2. Package SNU.geometryUtil contains 2 classes: IllegalTriangleException and Triangle.

Default Package contains Exception class which contains the main function for the program.

3. Default Package contains ExceptionPropogation class.

4. Package DynamicBinding contains 12 classes: FirstA, FirstB, FirstC, FirstD, SecondA, SecondB, SecondC, SecondD, ThirdA, ThirdB, ThirdC and ThirdD.

Default Package contains ImplementDynamicBinding class which contains the main function for the program.

**Input:**

1. Program prompts the user to enter the details related to every class.

2. Program asks the user to enter the dimensions for the triangle object created.

3. No input is required.

4. No input is required.

**Output:**

1. After every successive input of a class it's toString() function is called which displays all the data related to the class.

2. Program continues to print “No exception encountered” if the values entered are correct or else the program terminated throwing an exception.

3. Program outputs the source of the exception which describes the order of execution.

4. Program outputs the class from which the method is called.

**Remarks:**

Dynamic Binding is implemented here by creating several inheritance hierarchies. Every class contains a locateClass() function which returns a statement telling from where the function was called.

First(A,B,C,D) follow mutilevel inheritance where FirstA is the base class. An object of FirstD is created and is made to call locateClass() which through dynamic binding first searches D then C and then finally finds the implementation of the function in FirstB and so is called from FirstB.