**San José State University**
**Computer Science Department**
**CS151, Object Oriented Design and Programming, 05, Fall 2022**

## Homework #3

# Objective:

This homework's objective is to review and understand the units on exception handling and deep copy.

# Details:

Exercise 1:
Define and implement class **Course**. This class should contain the following fields: course name, course description, department, time the course starts, weekday the course is held on (for simplicity, let us assume the course only meets once a week). This class should contain getters and setters for all its attributes. This class also needs at least one constructor. Keep in mind that this class should be able to undergo deep copy (which interface does this class have to implement?). Save this class and its definition into a file named **Course.java**.

Define and implement class **Student**. This class should contain the following fields: first name, last name, age, gpa, major, department. Age should be an integer value. GPA should be a floating point value. This class should also contain one attribute field of type Course. You might declare it as:

*private Course course;*

Important note: For simplicity of this exercise we are declaring and using a single Course object. Since most students take more than one class we would normally declare a collection that holds any number of Course objects. We will be covering Java collections later in the semester.

Class Student should contain getters and setters for all its attributes. Implement a method called *printInfo()*, which will print the values of the Student instance to command line/terminal window (using built-in *System.out.println()* method). As was the case with the class Course, this class should be able to undergo deep copy. Save this class and its definition into a file named **Student.java**.

Define and implement class **StudentTest**. This class should implement *main()* method. In the body of the *main()* method you should create an instance of Student with the following information: John Smith, 20 year old, 3.6 gpa, Computer Science major, School of Computer Science department. This student will be taking a single course with the following details: CS151, Object Oriented Design and Programming, CS, 6:00pm, Tue.

Create another instance of Student class, which should be a clone (deep copy) of the first student. Use *printInfo()* method call to print the values of both instances of Student. Save this class and its definition into a file named **StudentTest.java**.

Exercise 2:
Define and implement class **Employee**. This class should contain the following fields: first name, last name, employee id, hourly pay. This class should contain getters and setters for all its attributes. This class also needs at least one constructor. Implement method *computePay()* in this class. This method should accept an integer number of hours the employee worked as an input argument and return a floating point value indicating the pay the employee earned. If the number of hours is invalid (e.g. a negative number) this method should throw *NumberFormatException*. If the number of hours is over 40 then this method should throw a custom exception named *TooManyHoursWorkedException*. Remember that you will have to define your own exception class for this. Save this class and its definition into a file named **Employee.java**.

Define and implement class **EmployeeTest**. This class should implement *main()* method. In the body of the *main()* method you should create an instance of Employee with the following information: John Smith, id = 101, hourly pay = $35/hr. Use *computePay()* method call to print to command line what this employee earned for the following number of hours:
40
23
1
0
-5
45

If an exception has been thrown by *computePay()* make sure to catch it and print the appropriate error message to command line. Save this class and its definition into a file named **EmployeeTest.java**.

# Submission:

Submit all files created by you for the homework exercises: Course.java, Student.java, StudentTest.java, Employee.java, TooManyHoursWorkedException.java, EmployeeTest.java and any other files you completed for this assignment, if any.

Make sure to submit by 11:59pm on the due date listed in Canvas. Submit your solution via Canvas.

If you have any questions, message me or the grader or both:
Yulia.Newton@sjsu.edu

madhujitaranjit.ambaskar@sjsu.edu

# Grading:

Your code must compile and execute successfully in order to get full credit for this assignment. For each exercise, I will compile and execute the files.

- Program with no compile errors
- Program executes
- Program outputs what is required by the exercise

A total of 20 points are possible for this homework assignment.