MPP Final Exam

26th April, 2017

Dr.Shaqfat Ali Shad

Problem 1. (10 points) Your package Problem1.exam contains two subpackages, **partA** and **partB**. Each contains a designated class file (respectively **PartA, PartB**), along with (possibly) other classes.

At the top of each of the designated class files, you will see a lambda expression. There are several things you will need to do with this expression.

a. Assign an appropriate type (some functional interface)
b. Express it as a method expression
c. State the type of method expression you have used
d. Express it as an inner class
e. Evaluate the lambda, the method expression and the inner class inside an **evaluator()** method.

There is a main method in each of the designated class files that attempts to run the evaluator method. In the body of the evaluator method, you should test your typed lambda expression, your method reference, and your inner class operation.

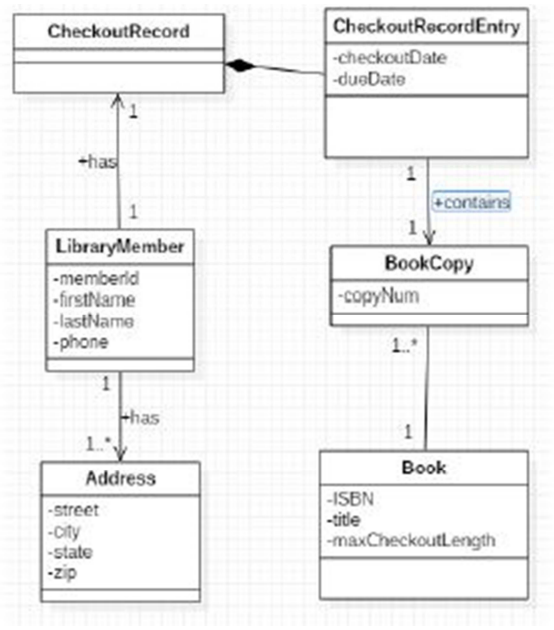
Each designated class provides a template for your work. You must follow this template. A sample solution is provided in the package Problem1.sample. Follow the format of this sample very closely.

The lambdas provided in the three parts are:

PartA: (Double x, Double y) -> x * y < x + y

PartB: (CheckoutRecord record) -> record.getCheckoutEntries()

Problem 2. (10 Points) Use the code in the **helperclasses** package and the class diagram below to help you solve the following problem.



Write a stream pipeline inside the main method of the class Problem2 (in package Problem2) that does the following: Print to the console the list of book titles –- in sorted order -- for which the book was checked out on June 27, 2015. The ordering of the book titles is as follows: First sort by the length of the title (number of characters), then by alphabetical order.

Use the data provided in the **TestData** class – a call for the list of **CheckoutRecordEntries** provided in that class has already been made for you in the main method – use the list provided there.

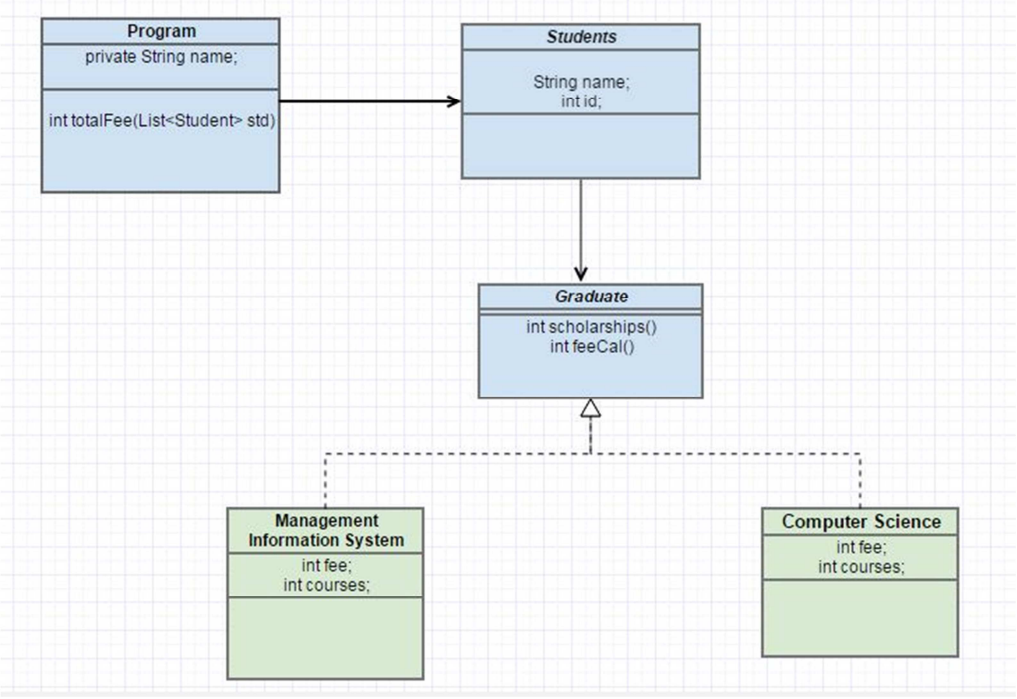
Problem 3. (10 points) In your package prob3.exam there are three classes: Person, **FindOldestPerson, TestCode**. The **FindOldestPerson** class contains an unimplemented static method **findOldestPerson(List<Person> list).** For this problem implement this method, which must return the oldest Person object occurring in the input list. Your implementation must use the reduce method for Streams. (If you do not use the **reduce** method, you will receive less than 50% credit for this problem.)

The class **TestCode** provides test data to test your method. You may use this class, but do not modify it in any way.

Problem 4. (10 points) High school students at Fairfield High School participate in an annual word contest. Students are given 15 minutes to type into a test computer as many words as they can think of that do not begin with one of the illegal letters. In this year's contest, the illegal letters are A, B, C, E, M, N, R, S, T. The evaluator program for student submissions runs a method **adjustWords** which takes the list of words created by a participant, checks that all the words are legal, turns them into lower case words, and produces a new list with these modified words. The code for this is shown in the class **WordGame** in your Problem4.exam package. However, it has been commented out because it is implemented as a lambda pipeline in which one of the lambda implementations throws an exception which is not supported by the interface type of the lambda.

Modify the implementation of this method so that the exception is handled in one of the standard ways. Then make sure that the main method executes test data as expected (you will need to uncomment the code in the main method).

Problem 5. (10 Points)



UML diagram shows relationship among classes, shell of these classes is given in Problem5.Exam folder. Garduate interface has two functions int scholarships() and int fecal(), where int scholarship() returns scholarship amount for both CS and MIS @ fixed 2500 USD which is non changeable. While int feecal() by default calculates it by formula fee*courses+1000, otherwise for CS it is fee*courses+1500 and for MIS it is fee*courses+1700.

Uncomment code given in the main() available in Program class, you cannot change code in main() as it creates type of Graduate student i.e. Computer Science or Management information Science. You are not supposed to add any further classes beside given in UML, you can add functions in shell as per requirement or can change type of functions as per requirement (instance/static).

Problem 6. (10 points) In Problem6.Exam you have one file **MaximumTest** class, it has only one method **public static String maximum(String x, String y, String z)** which returns maximum of the three values sent to it. You need to convert this method into generic so that it can work for all other values commented out in main() method.

Problem7. (5Points) Write a paragraph that link any of the topic we studied during MPP course with SCI.