Computer Programming II (CS141)
Assignment 1

- ***Deadline: Saturday 20/02/2016 at 11:59 pm***
- ***Submission has to be made via the submission folder assigned by your Instructor on Blackboard.***
- ***No Email Submission will be accepted.***
- ***Late submission will result in ZERO marks.***
- ***Identical work will result in ZERO marks.***
- ***This Assignment worth 5 marks.***

Q1. In the waterfall model and a programmer at the implementation stage noticed the design need to be changed to have high performance software. Would it be possible to change or add some specifications, justify your answer?

(0.5 Marks)

Q2. Can you convert a superclass reference into a subclass reference? A subclass reference into a superclass reference? If so, give examples. If not, explain why not.

(0.5 Marks)

Q3. Draw the state machine diagram for the following scenario?

Consider a copy of book in a library that can go through several states: available, on loan, reserved, etc. Suppose, for simplicity, we put reservations on a copy rather than a book.

(1 Mark)

Q4. Write the code to declare the trivial class inside a method?

(1 Mark)

Q5. Consider the following problem description and write java code.

(2 Marks)

A. Consider a superclass PurchaseItem which models customer's purchases. This class has:
   1. Two private instance variables name and unit price.
   2. One constructor to initialize the instance variables.
   3. A default constructor to initialize name to "no item", and unit price to 0.
   4. A method that returns the price.
   5. A toString method to return the name of the item followed by @ symbol, then the unit price.

B. Consider two subclasses WeighedItem and CountedItem. WeighedItem has an additional instance variable weight while CountedItem has an additional variable quantity.

   1. Write an appropriate constructor for each of the classes making use of the constructor of the superclass in defining those of the subclasses.
   2. Override getPrice method that returns the price of the purchasedItem based on its unit price and weight (WeighedItem), or quantity (CountedItem). Make use of getPrice of the superclass
   3. Override also toString method for each class making use of the toString method of the superclass in defining those of the subclasses.
   4. toString should return something that can be printed on the receipt.
      For example
      Banana @ 3.00   1.37Kg   4.11 SR (in case of WeighedItem class)
      Pens @ 4.5   10 units   45 SR (in case of CountedItem class)

C. Write an application class where you construct objects from the two subclasses and print them on the screen.