# CSC 2310 – Spring 2016
# Homework #3
# Due 26/2/2016 11:30 pm

***Submission Requirements***
*You must turn work at the SPECIFIED TIME so you can receive credit for Homework! Homework 2 part a must be completed in myprogramminglab.com and part b* ***must be submitted on brightspace*** *by the due date and time. Late homework will be subject to a penalty, as stated in the course grading policy. No email or hard copies of homework will be accepted.*

*You may discuss the assignments with other students in the class, but (as stated in the academic honesty policy) your written answers* ***must be your own****, and you must list the names of other students you discussed the assignment with.*

**How to Submit**

*This homework has two parts - one part must be done in* myprogramminglab.com *- the other part must be done in Eclipse and your .java and .class files uploaded to brightspace. Please follow the directions carefully for each part.*
*This homework is worth 2 individual grades so be prepared to spend some time on it!*

***Part a - MPL***
Do all the questions from section 10.1 and 10.2. Be sure to submit each of your answers. You get 3 tries, so if your answer is incorrect, you can still try again.
There are 25 questions in these sections so do not wait until the last minute!

***Part b - Eclipse/Java***

1. Write a java program that calls a method called reverse3 that accepts an ArrayList of integer values as a parameter and reverses each successive sequence of three values in the list. If the list has extra values that are not part of a sequence of three, those values are unchanged. For example if a list stores values [10, 13, 2, 8, 7, 90, -1, 2], after the call the list should store the values [2, 13, 10, 90, 7, 8, -1, 2]. The first sequence of three (10, 13, 2) has been reversed to (2, 13, 10). The second sequence (8, 7, 90) has been reversed to (90, 7, 8) and so on. Notice that -1 and 2 are unchanged because they were not part of a sequence of three values. Print the array before the call and after the call.

2. Write a class called Date that includes three fields year, month and day. This class stores information about a single date (year, month and day). Your class should have constructor(s), accessors and mutators and it should implement the Comparable interface. Years take precedence over months, which take precedence over days. For example, Feb 19, 2016 comes after Nov 20, 2015.

The following class DateTest can be used to test the Date class that you wrote. It creates a list of the birthdays of the first 5 U.S. presidents in random order and puts them into sorted order. (Note: you can use Collections.sort() to sort your ArrayList after you implement the compareTo() method).

```
import java.util.*;

public class DateTest {
   public static void main(String[] args) {
      ArrayList<Date> dates = new ArrayList<Date>();
      dates.add(new Date(4, 13, 1743));   // Jefferson
      dates.add(new Date(2, 22, 1732));   // Washington
      dates.add(new Date(3, 16, 1751));   // Madison
      dates.add(new Date(10, 30, 1735));  // Adams
      dates.add(new Date(4, 28, 1758));   // Monroe

      System.out.println("birthdays = " + dates);
      Collections.sort(dates);
      System.out.println("birthdays = " + dates);
   }
}
```

When you execute the following code it should print:
birthdays = [4/13/1743, 2/22/1732, 3/16/1751, 10/30/1735, 4/28/1758]
birthdays = [2/22/1732, 10/30/1735, 4/13/1743, 3/16/1751, 4/28/1758]