# ICS 211 Fall 2014 Exam 2, Nov 12th, 2014

Clearly write your name on both the **back** and **front** of this exam.

This exam is closed-book. No calculators are allowed. There are a total of 100 points.

Be sure to answer all parts of each question.

**Question 1** (10 points): Write a **recursive** method **even** that takes as a parameter a **LinkedNode<E>** and returns a **boolean**. If the parameter is null or refers to a linked list of even length, the method should return **true**. Otherwise the method should return **false**.

**Question 2** (5 points): Write the **Stack<E>** interface.

**Question 3** (10 points): Implement the following method.

**public class ArrayStack<E> implements StackInterface<E> {**

**private int top;**

**private E[] array;**

**public E peek() throws EmptyStackException {**

**Question 4** (20 points): Implement this method to perform a binary search. If the value is found, the method returns the index of the value, otherwise the method returns -1. The array **data** is sorted in ascending order. You may write a helper method.

**static int binarySearch(java.lang.Comparable value, Object[] data) {**

**Question 5** (10 points): Create a max heap (stored in an array) by inserting, in sequence, the values 7, 1, 4, 5, 6, 3, 8, 2. Show the resulting array.

**Question 6** (5 points): Which of the following is a postorder traversal of a binary tree?

1. Visit root node, traverse TR, traverse TL
2. Traverse TL, traverse TR, visit root node
3. Visit root node, traverse TL, traverse TR
4. Traverse TL, visit root node, traverse TR

**Question 7** (5 points): What is the worst-case runtime of binary search on a sorted array of *n* items.

**Question 8** (5 points): Fill in the blank for both of these sentences.

The node at the top of a tree is called its \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Nodes that have the same parent are \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**Question 9** (5 points): Write the **Queue<E>** interface.

**Question 10** (5 points): What is the worst-case runtime to add a value to a heap that has *n* values?

**Question 11** (5 points): Briefly, what is the difference between a binary search and a binary search tree?

**Question 12** (5 points): What is the worst-case runtime to add a value to a binary search tree with *n* values?

**Question 13** (10 points): Implement the following method.

**public class ArrayQueue<E> implements Queue<E> {**

**private int front;**

**private int rear;**

**private E[] array;**

**public E remove() throws NoSuchElementException {**