Tutorial 11 Solutions

All code produced using Java™ software.

Tutorial Problems

Problem 1

a. False. In an insertion sort, items are selected one at a time from an unsorted set and inserted in a sorted list.
b. True
c. True
d. False. Some Collection implementations are unordered; some are ordered.
e. True

Problem 2

Part A

Given a list as follows
10 20 30 40 25 35
consider the Insertion Sort algorithm discussed in class

1. How many comparisons would this array require the Insertion Sort to perform a complete sorting?

8

2. Show the state of the array after each comparison

10 20 30 40 25 35
10 20 30 40 25 35
10 20 30 40 25 35
10 20 25 30 40 35
10 20 25 30 40 35
10 15 20 25 35 40
10 15 20 25 35 40

3. We have two loops in this Insertion Sort method, in your answer to (2), mark the state of the array at the end of each iteration of the outermost loop. (3-1) => (3-5)

Part B

Simulate the execution of quicksort on the following array, using the simpler Quick Sort algorithm discussed in class to get the first two subarrays:

6 12 18 9 15 3
Pivot = 6
6 12 18 9 15 3
L H
3 12 18 9 15 6
L H
3 12 18 9 15 6
H L

Two subarrays are: {3} and {12, 18, 9, 15, 6}

Problem 3
Design Exercises

Part 1.
A.
Answer: b.
B.
Using Student's IDs as the key, one only needs to perform O(log n) number of comparisons to find a particular student's grade.
Using Student's grade as the key, one needs to perform O(n) number of comparisons to find a particular student's grade, since we must search through the entire tree to find a student with the matching ID.

Part 2.
We will use students' IDs as keys to both hash tables and binary search trees. Hashtables will give the best lookup performance (constant lookup time), assuming that student IDs are uniformly distributed.

Part 3.

Answer: b.

Part 4.

Part 5.
A.

```java
public class Student {
    String name;
    int id;
    int grade;
    public int hashCode() {
        return id;
    }
    public boolean equals (Object s) {
        //insert your code below
        Student stu = (Student) s;
        if (stu.id == id) {
            return true;
        } else {
            return false;
        }
    }
}
```

B.

Answer: 100, 200, 300, 400, 500