Computing & Algorithms II

CPSC-102 **Summer 1999**
Final Exam **30 September 1999**

Instructions: This exam is closed-book, closed-notes, and is to be completed individually. Complete all problems; each problem indicates the number of points possible. There are 150 total points. Show your work; partial credit may be awarded.

1. *5 points each.* Describe the role played by each of the following classes or concepts in a Java graphical application.

   (a) Listener

   (b) Panel

   (c) ItemEvent

   (d) CheckboxGroup

   (e) Menu

   (f) Interface

2. *10 points each. 10 points each.* For each of the following pairs of items, list an advantage which each has over the other. (E.g. "Pepsi tastes better, but Coke is cheaper.")

   (a) Heaps versus unsorted arrays as implementations for a priority queue

   (b) Linked lists versus binary search trees

   (c) Mergesort versus Quicksort

3. *10 points each.* We have discussed three general categories of sorting algorithms in class: priority queue sorts, divide & conquer sorts, and non-comparison-based sorts. For each of the following sorts, give a *brief* description of the algorithm, name the category above to which it belongs, and give its average case runtime. (Hint: all sorting algorithms we have considered have been either $O(n)$, $O(n \log_2 n)$, or $O(n^2)$.)

   (a) Quicksort

   (b) Radix Sort

   (c) Bubble Sort

   (d) Heap Sort

4. Consider the binary search tree data structure discussed and used throughout this portion of the course.

   (a) *10 points.* Give a class definition appropriate for a binary search tree containing integers.

   (b) *20 points.* Using your definition above, write a method which computes the number of nodes in such a tree.

   (c) *20 points.* Using your definition above, write a method `itemAt()` which takes an integer n and returns the item which would appear at position n in a sorted listing of the elements of that tree.