

### Masters Exam, Algorithms: Spring 2000

Answer a total of four problems, two from the easy group, and two from the hard group.

#### Easy

1. Give a definition and draw a graph to illustrate each of the following.
  - $O(f(n))$
  - $\Theta(f(n))$
  - $\Omega(f(n))$
2. We're in Las Vegas, using a slot machine. It has three wheels, each with 8 different values (Elvis, Horseshoe, 7, Dollar Sign, Hearts, Clubs, Spades, Diamonds). The machine takes one quarter, and pays out one quarter if two of the values match, and one dollar if all three match. What are the odds that we'll get a quarter back? What are the odds that we'll get a dollar back?
3. We implement a simple linked list, in which we append at the end of the list. To perform the append, we traverse the list to find the last element, then attach the value. What is the complexity of inserting  $n$  variables, and how can we improve this?
4. Give the Big-O for good implementations of algorithms on these common problems.
  - Insert of an element into a balanced binary tree.
  - Breadth-first search of a graph.
  - Sorting an array of  $n$  integers.

#### Hard

1. Sketch pseudocode for Dijkstra's shortest path algorithm. Be sure to state clearly what sort of data structures you use. How is Dijkstra's algorithm different from ordinary breadth first search?
2. Sketch pseudocode for Prim's or Kruskal's spanning tree algorithm. State which one you're using.
3. Elvis has returned from the grave, and is planning a tour. He wants to play a concert at the capital of each state, and he absolutely wants to spend the least amount of time possible travelling. He has asked you to plan the tour; what do you tell him? (Yes, there is an appropriate computer-science related answer.)
4. We have a set of tasks, and a set of employees. Each employee can handle a few tasks, but not necessarily all of them. Each employee can be assigned one and only one task, and each task requires one and only one employee. How would we determine an assignment of employees to tasks such that we handle the maximum number of tasks possible? It is sufficient to explain how to apply a well known algorithm to this problem.
5. We have a small gift for Don Knuth on his birthday. Unwrapping a box is fun; unwrapping a box inside a box is even more fun. If we have a variety of boxes, describe an algorithm to maximize the number of boxes we use to wrap the present.

For example, if we have  $3 \times 3 \times 3$ ,  $4 \times 7 \times 8$ , and  $6 \times 2 \times 6$  boxes, we can place the  $3 \times 3 \times 3$  box into the  $4 \times 7 \times 8$  box, and use two boxes to wrap the present. The algorithm should work for an arbitrary number of boxes, find the maximum number of boxes, and be reasonably efficient. A box will fit into another box if at least one of the three dimensions is larger; you can rotate and turn boxes any way you want to.