PhD Exam – Spring 2004 – Algorithms
Answer as much of the seven questions as you can.

1. Give formal definitions for O, Ω, Θ, P, and NP.

2. Suppose we are travelling internationally, and want to bring some items through customs. There is a restriction on the total weight, the total volume, and the total cost of the items; item i has weight w_i, volume v_i, and cost c_i. Describe an algorithm that will find a set of items to maximize the total cost of the items we bring, subject to a total weight restriction of W and a total volume restriction of V. You can assume that the weights, volumes, and costs of the items are relatively small integers, and there are a limited number of items that we are considering.

3. Describe extended heaps, and how they are used in Dijkstra's algorithm.

4. Disjoint set operations are used in Kruskal's algorithm; describe the basic algorithms, and in particular, detail the use of path compression.

5. The Ford-Fulkerson method has a weakness, that is addressed by the Edmonds-Karp algorithm. Describe this weakness.

6. Randomization is used in some quicksort implementations, to address algorithm complexity issues. Why would we use randomization? What is the "expected" Big-O complexity from a randomized implementation? What is the "not unlikely" Big-O complexity if we don't use randomization?

7. Suppose we have a round table, and a group of people, some of whom know eachother. We wish to have the people seated at the table such that everyone knows the two people they are seated next to. Is it possible to have an efficient algorithm to find a seating plan? If so, describe it. In not, why?