Each problem is worth 10 points. We suggest the following steps in writing up your solutions, when they are applicable:

1. Understand the problem. Take a small numeric example and work it out manually to get a clear idea about the problem. Do not hesitate to contact us if you do not understand what you are supposed to do.

2. Give an informal description (at most 10 lines of English text) of your solution. The informal description should highlight the key ideas of your solution. Your solution may just be graded by the quality of this informal description.

3. Write up your design; Use the pseudo programming language from the book to write your algorithms. Write your algorithms modularly. Provide sufficient detail as warranted by the problem.

4. Analyze the time complexity.

5. Argue that your algorithm is correct.

**Problem 1: (Function Order)**
Is the function $|\log n|!$ polynomially bounded? Is the function $|\log \log n|!$ polynomially bounded?

**Problem 2: (Recurrence)**
The running time of an algorithm $A$ is described by the recurrence $T(n) = 7T(n/2) + n^2$. A competing algorithm has a running time of $T'(n) = aT'(n/4) + n^2$. What is the largest integer value for $a$ such that $A'$ is asymptotically faster than $A$?

**Problem 3: (Sorting)**
Show how to sort $n$ integers in the range 1 to $n^2$ in $O(n)$ time.

**Problem 4: (2-4, Skiena)**
The mode of a set of numbers is the number that occurs most frequently in the set. The set (4,6,2,4,3,1) has a mode of 4.

1. Give an efficient and correct algorithm to compute the mode of a set of $n$ numbers;
2. Suppose we know that there is an (unknown) element that occurs $n/2 + 1$ times in the set. Give a worst-case linear-time algorithm to find the mode. For partial credit, your algorithm may run in expected linear time.