Assignment 1

due Friday, October 6, 2006

1. Suppose that algorithm \( A \) uses \( 1250n^2 \) operations while algorithm \( B \) uses \( 2n^4 \) operations. Determine the value \( n_0 \) such that \( A \) is as fast or faster than \( B \) for all \( n \geq n_0 \). [4 points]

2. exercise 3.1-4, p 50. Additionally, is \( 2^{2n+1} = O(2^{2^n}) \)? [4 points]

3. exercise 3.2, p 58 [8 points]

4. exercise 3-3, part a (not part b), p 58. [8 points]

5. An algorithm takes \( 0.2\mu \) for input size 100 (this allows you to determine the constant, which will be different in each case). How large of an input can be solved in 5 minutes if the algorithm is . . . ?
   (a) \( \Theta(n) \)
   (b) \( \Theta(n \log n) \)
   (c) \( \Theta(n^3) \)
   (d) \( \Theta(2^n) \)

   [8 points]

6. Describe how to find the minimum and maximum of an array of \( n \) elements with at most \( \frac{3}{2}n \) element comparisons. (Do not count comparisons needed for the array indices.) [4 points]

Total: 36 points

Notes:

- A \( \mu \) is 1/1000 of a second.
- For \( \Theta(n \log n) \), you may find an approximate solution, but try to be as accurate as you can stand.
- Hint for Q6: form \( \lceil \frac{n}{7} \rceil \) pairs, from each pair find candidate min and candidate max for the whole list.