Assignment 1
CSCI 4520, Fall 2003
Due Wednesday, September 17, 10:00am

Please, submit your solutions using WebCT.

1. [*4 points*] For the following pairs of functions f, g determine whether

- **i.** $f(n) \in O(g(n))$,
- **ii.** $f(n) \in \Theta(g(n))$,
- **iii.** $f(n) \in \Omega(g(n))$,
- **iv.** $f(n) \in o(g(n))$,
- **v.** $f(n) \in \omega(g(n))$.

Justify all your answers (including negative ones).

- **a)** $f(n) = n \log n$, $g(n) = n^2$
- **b)** $f(n) = \sum_{i=1}^{n} i$, $g(n) = n^2$
- **c)** $f(n) = n^{10}$, $g(n) = 2^n$
- **d)** $f(n) = 2^n$, $g(n) = n!$

2. [*4 points*] In this question, assume that f, g are functions mapping positive integers to positive reals. Prove or disprove the following statements.

- **a)** $O(log_a n) = O(log_b n)$, for all constants $a, b \geq 0$.
- **b)** $O(n^k) \subseteq O(n^{k+1})$, for $k > 0$.
- **c)** $\Theta(c \times f(n)) = \Theta(d \times f(n))$, for constants c and d.
- **d)** $g(n) \in \Omega(f(n))$, iff $f(n) \in O(g(n))$.

3. [*12 points*] **Implementation and Analysis of Algorithms.** Please implement the *sequential search, binary search* and *bubble sorting* algorithms given in the slides in C++ or Java.

Inputs. The elements in the list are assumed to be real numbers, and should be generated randomly. For the two search algorithms, the element to be searched should also be generated randomly, with the probability that the element exists in the list being 0.5.

Execution Time. Run your programs with different input sizes and record the execution time for each run. For the two search algorithms, each input size should run 6 times, and each time with a different element to search. Then the average time of the 6 runs can be calculated for each input size. Specifically, the following input sizes should be tested for each algorithm: 10000, 50000, 100000, 500000, 1000000. For each of the two search algorithms, the total number of runs is then $5 \times 6 = 30$. Notice

that for the binary search, the time that is required to sort the list should not be counted in the execution time for searching.

Plot. Plot a graph like Figure 1.3 on page 28 in the textbook. The X-coordinate is the input size, and the Y-coordinate is the execution time. For the two search algorithms, the execution time for each input should be the average of the 6 runs. The graph can be plotted by GNUPLOT with high quality, and any other appropriate software can also be used.

Report and Analysis. Write a brief report documenting what you did and how you did in the experiments, e.g., the machine and the operating system that your experiements are run, how the input data are generated, how the execution time is recorded, etc. In the report, you are also required to analyz the plotted graph, and compare it to the complexity functions for these algorithms, e.g., is the graph consistent with the complexity functions? and if not, analyze the reasons. The report should not exceed 2 pages including the graph.