Solutions: Mock Exam for Midterm II
Discrete Mathematical Structures CS3233

November, 2005

1. Define $f(n) = \mathcal{O}(g(n))$, $f(n) = \Omega(g(n))$, and $f(n) = \Theta(g(n))$.
   Solution: Refer to Section 3.2.

2. Prove or disprove: $(n^2 + n^3)/2 = \Theta(n^3)$.
   Solution: We must show that $(n^2 + n^3)/2 = \mathcal{O}(n^3)$ and $(n^2 + n^3)/2 = \Omega(n^3)$. Taking $n > 1$, we have $(n^2 + n^3)/2 \leq n^3$ and $(n^2 + n^3)/2 \geq n^3/2$, which demonstrate the respective conditions.

3. What is the best big-O estimate of the number of comparisons that are performed by an algorithm that takes a list of n integers and finds the least of the first 100 values? Justify your answer.
   Solution: $\mathcal{O}(1)$. See discussion of Table 1 on page 148 in the text.

4. What is the expected-case complexity of finding the least value in a list of n integers? Select the one best answer from the following list: $\mathcal{O}(1), \mathcal{O}(\log n), \mathcal{O}(n), \mathcal{O}(n \log n), \mathcal{O}(n^2), \mathcal{O}(n^3), \mathcal{O}(2^n)$?
   Solution: $\mathcal{O}(n)$. It is not possible to find the least value without looking at all the values: if an algorithm skipped any value and it happened to be the least one, the algorithm would be wrong.

5. How is the complexity of a problem defined?
   Solution: See slide 4 of lecture notes for 10/21/05.

6. What are P and NP?
   Solution: See slide 7 of lecture notes for 10/21/05.

7. Can an NP-Complete problem be solved by a (deterministic) algorithm having polynomial complexity?
   Solution: Saying that the algorithm is deterministic just emphasizes that it has none of the guess operations that characterize non-deterministic computation. This is what is normally meant by the term, algorithm.

   There is no known proof that enables us to answer this question conclusively either way. NP-Complete problems have the property that given an oracle that solves such a problem, any other NP problem can be solved by deterministic procedure that runs in polynomial time. Thus if any NP-complete problem can be solved in polynomial time, so can all problems in NP. However, no polynomial-time algorithm is known for any NP-complete problem.

8. Suppose the integer m ends with 10 zeros. What can you conclude about m's prime factorization?
   Solution: The prime factorization of m must contain at least 10 2's and 10 5's, since 10^{10} divides m.

9. Let a and b be positive integers, let $p_1, p_2, \ldots, p_n$ enumerate the primes less than $max(a, b)$ in increasing order, and let $a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n}$ and $b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$ be the prime factorizations of a and b.

   Prove that $\text{lcm}(a, b) = p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \cdots p_n^{\max(a_n,b_n)}$ by showing: (a) That both a and b divide $p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \cdots p_n^{\max(a_n,b_n)}$, and by showing (b) That no smaller number is divisible by both a and b.

   Solution: Part (a) holds because each prime factor of a occurs at least as many times in $p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \cdots p_n^{\max(a_n,b_n)}$. The same is true of each prime factor of b.

   For part (b), suppose for contradiction that k is smaller and is divisible by both a and b. The prime factorization of $k = p_1^{k_1} p_2^{k_2} \cdots p_m^{k_m}$ must have $k_i < \max(a_i, b_i)$ for some i, since otherwise k is not smaller. But this means that $k_i < a_i$ or $k_i < b_i$. When $k_i < a_i$, it is not possible that a divides k. Similarly when $k_i < b_i$, it is not possible that b divides k. This gives us the desired contradiction.

10. Given positive integers a and b, prove that $\gcd(a, b) \cdot \text{lcm}(a, b) = ab$.
Solution:

$$\begin{aligned} ab &= p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n} \\ &= p_1^{\min(a_1,b_1)} p_2^{\min(a_2,b_2)} \cdots p_n^{\min(a_n,b_n)} p_1^{\max(a_1,b_1)} p_2^{\max(a_2,b_2)} \cdots p_n^{\max(a_n,b_n)} \\ &= \gcd(a, b) \cdot \text{lcm}(a, b) \end{aligned}$$

11. Let m be a positive integer. Show that if $a \equiv b \pmod{m}$ and $c \equiv d \pmod{m}$, then $a + c \equiv b + d \pmod{m}$ and $ac \equiv bd \pmod{m}$.
Solution: See proof of Theorem 10 on page 163.

12. Use induction to show that $P(n) \equiv \sum_{1 \le i \le n} (-1)^{i-1} i^2 = (-1)^{n-1} n(n + 1)/2$ holds for all positive integers n. **Solution:** See solution to problem 17 section 3.3 on Page S-25.

13. Is the set of positive rational numbers well ordered? Why or why not?
Solution: It is not well ordered because if we consider the whole set, there is no least element. To see this, consider any positive rational q; $q/2$ is also rational and $q/2 < q$.

14. Is the set of negative integers well ordered? Why or why not?
Solution: It is not because any subset containing an infinite number of negative integers has no least element.

15. Is the set of integers greater than 100 well ordered? Why or why not?
Solution: It is because any subset of this set is also a subset of N, so it has a least element, since N is well ordered.