

# CMPS 101

## Midterm 1

### Review Problems

**Problem 19 in this handout may not be included. It depends on how far we get by Wednesday.**

1. Let  $f(n)$  and  $g(n)$  be asymptotically non-negative functions which are defined on the positive integers.
  - a. State the definition of  $f(n) = O(g(n))$ .
  - b. State the definition of  $f(n) = \omega(g(n))$
2. State whether the following assertions are true or false. If any statements are false, give a related statement that is true.
  - a.  $f(n) = O(g(n))$  implies  $f(n) = o(g(n))$ .
  - b.  $f(n) = O(g(n))$  if and only if  $g(n) = \Omega(f(n))$ .
  - c.  $f(n) = \Theta(g(n))$  if and only if  $\lim_{n \rightarrow \infty} (f(n)/g(n)) = L$ , where  $0 < L < \infty$ .
3. Prove that  $\Theta(f(n)) \cdot \Theta(g(n)) = \Theta(f(n) \cdot g(n))$ . In other words, if  $h_1(n) = \Theta(f(n))$  and  $h_2(n) = \Theta(g(n))$ , then  $h_1(n) \cdot h_2(n) = \Theta(f(n) \cdot g(n))$ .
4. Let  $f(n)$  and  $g(n)$  be asymptotically positive functions (i.e.  $f(n) > 0$  and  $g(n) > 0$  for all sufficiently large  $n$ ), and suppose  $f(n) = \Theta(g(n))$ . Does it necessarily follow that  $\frac{1}{f(n)} = \Theta\left(\frac{1}{g(n)}\right)$ ? Either prove this statement, or give a counter-example.
5. Give an example of functions  $f(n)$  and  $g(n)$  such that  $f(n) = o(g(n))$  but  $\log(f(n)) \neq o(\log(g(n)))$ . (Hint: Consider  $n!$  and  $n^n$  and use the corollary to Stirling's formula in the handout on common functions.)
6. Let  $g(n)$  be an asymptotically non-negative function. Prove that  $o(g(n)) \cap \Omega(g(n)) = \emptyset$ .
7. Use limits to prove the following (these are some of the exercises at the end of the asymptotic growth rates handout):
  - a. If  $P(n)$  is a polynomial of degree  $k \geq 0$ , then  $P(n) = \Theta(n^k)$ .
  - b. For any positive real numbers  $\alpha$  and  $\beta$ :  $n^\alpha = o(n^\beta)$  iff  $\alpha < \beta$ ,  $n^\alpha = \Theta(n^\beta)$  iff  $\alpha = \beta$ , and  $n^\alpha = \omega(n^\beta)$  iff  $\alpha > \beta$ .
  - c. For any positive real numbers  $a$  and  $b$ :  $a^n = o(b^n)$  iff  $a < b$ ,  $a^n = \Theta(b^n)$  iff  $a = b$ , and  $a^n = \omega(b^n)$  iff  $a > b$ .
  - d.  $f(n) + o(f(n)) = \Theta(f(n))$ .

8. Let  $g(n) = n$  and  $f(n) = n + \frac{1}{2}n^2(\sin(n) + 1)$ . Show that

a.  $f(n) = \Omega(g(n))$

b.  $f(n) \neq O(g(n))$

c.  $\lim_{n \rightarrow \infty} \left( \frac{f(n)}{g(n)} \right)$  does not exist, even in the sense of being infinite.

Note: this is the ‘Example C’ mentioned in the handout on asymptotic growth rates.

9. Use Stirling’s formula:  $n! = \sqrt{2\pi n} \cdot \left(\frac{n}{e}\right)^n \cdot (1 + \Theta(1/n))$ , to prove that  $\log(n!) = \Theta(n \log n)$ .

10. Use Stirling’s formula to prove that  $\binom{2n}{n} = \Theta\left(\frac{4^n}{\sqrt{n}}\right)$ .

11. Consider the following *sketch* of an algorithm called ProcessArray which performs some unspecified operation on a subarray  $A[p \cdots r]$ .

ProcessArray(A, p, r) (Preconditions:  $1 \leq p$  and  $r \leq \text{length}[A]$  )

1. Perform 1 basic operation

2. if  $p < r$

3.  $q \leftarrow \left\lfloor \frac{p+r}{2} \right\rfloor$

4. ProcessArray(A, p, q)

5. ProcessArray(A, q+1, r)

a. Write a recurrence formula for the number  $T(n)$  of basic operations performed by this algorithm when called on the full array  $A[1 \cdots n]$ , i.e. by ProcessArray(A, 1, n). (Hint: recall our analysis of MergeSort.)

b. Show that the solution to this recurrence is  $T(n) = 2n - 1$ , whence  $T(n) = \Theta(n)$ .

12. Consider the following algorithm which does nothing but waste time:

WasteTime(n) (pre:  $n \geq 1$ )

1. if  $n > 1$

2. for  $i \leftarrow 1$  to  $n^3$

3. waste 2 units of time

4. for  $i \leftarrow 1$  to 7

5. WasteTime( $\lceil n/2 \rceil$ )

6. waste 3 units of time

a. Write a recurrence formula which gives the amount of time  $T(n)$  wasted by this algorithm.

b. Show that when  $n$  is an exact power of 2, the solution to this recurrence relation is given by

$$T(n) = 16n^3 - \frac{1}{2} - \frac{31}{2}n^{\lg 7}, \text{ and hence } T(n) = \Theta(n^3).$$

13. Define  $T(n)$  by the recurrence formula

$$T(n) = \begin{cases} 1 & 1 \leq n < 3 \\ 2T(\lfloor n/3 \rfloor) + 4n & n \geq 3 \end{cases}$$

Use Induction to show that  $\forall n \geq 1: T(n) \leq 12n$ , and hence  $T(n) = O(n)$ .

14. Prove that all trees on  $n$  vertices have  $n-1$  edges. Do this by (a) induction on the number of vertices, and (b) by induction on the number of edges.

15. Define  $S(n)$  for  $n \in \mathbb{Z}^+$  by the recurrence:

$$S(n) = \begin{cases} 0 & \text{if } n = 1 \\ S(\lceil n/2 \rceil) + 1 & \text{if } n \geq 2 \end{cases}$$

Use induction to prove that  $S(n) \geq \lg(n)$  for all  $n \geq 1$ , and hence  $S(n) = \Omega(\lg n)$ .

16. Let  $f(n)$  be a positive, increasing function that satisfies  $f(n/2) = \Theta(f(n))$ . Show that

$$\sum_{i=1}^n f(i) = \Theta(nf(n))$$

(Hint: Emulate the **Example** on page 4 of the handout on asymptotic growth rates in which it is proved that  $\sum_{i=1}^n i^k = \Theta(n^{k+1})$  for any positive integer  $k$ .)

17. Use the result of the preceding problem to give an alternate proof of  $\log(n!) = \Theta(n \log(n))$  that does not use Stirling's formula.

18. Let  $T(n)$  be defined by the recurrence formula

$$T(n) = \begin{cases} 1 & n = 1 \\ T(\lfloor n/2 \rfloor) + n^2 & n \geq 2 \end{cases}$$

Show that  $\forall n \geq 1: T(n) \leq \frac{4}{3}n^2$ , and hence  $T(n) = O(n^2)$ .

\*\*\*\*\*

We may not get far enough for this problem. If we do I'll let you know.

19. Define  $T(n)$  by the recurrence formula:

$$T(n) = \begin{cases} 7 & 1 \leq n < 3 \\ 2T(\lfloor n/3 \rfloor) + 5 & n \geq 3 \end{cases}$$

- Use the iteration method to determine a solution to the above recurrence.
- Use the solution you found in part (a) to determine an asymptotic solution.