

elsewhere for more depth. Algorithm texts more heavily stressing analysis include [CLRS01, KT06].

The book *Concrete Mathematics* by Knuth, Graham, and Patashnik [GKP89] offers an interesting and thorough presentation of mathematics for the analysis of algorithms. Niven and Zuckerman [NZ80] is an nice introduction to number theory, including Waring’s problem, discussed in the war story.

The notion of dominance also gives rise to the “Little Oh” notation. We say that $f(n) = o(g(n))$ iff $g(n)$ dominates $f(n)$. Among other things, the Little Oh proves useful for asking questions. Asking for an $o(n^2)$ algorithm means you want one that is better than quadratic in the worst case—and means you would be willing to settle for $O(n^{1.999} \log^2 n)$.

2.10 Exercises

Program Analysis

- 2-1. [3] What value is returned by the following function? Express your answer as a function of n . Give the worst-case running time using the Big Oh notation.

```
function mystery(n)
  r := 0
  for i := 1 to n - 1 do
    for j := i + 1 to n do
      for k := 1 to j do
        r := r + 1
  return(r)
```

- 2-2. [3] What value is returned by the following function? Express your answer as a function of n . Give the worst-case running time using Big Oh notation.

```
function pesky(n)
  r := 0
  for i := 1 to n do
    for j := 1 to i do
      for k := j to i + j do
        r := r + 1
  return(r)
```

- 2-3. [5] What value is returned by the following function? Express your answer as a function of n . Give the worst-case running time using Big Oh notation.

```
function prestiferous(n)
  r := 0
  for i := 1 to n do
    for j := 1 to i do
      for k := j to i + j do
        for l := 1 to i + j - k do
```

```
     $r := r + 1$   
  return( $r$ )
```

- 2-4. [8] What value is returned by the following function? Express your answer as a function of n . Give the worst-case running time using Big Oh notation.

```
function conundrum( $n$ )  
   $r := 0$   
  for  $i := 1$  to  $n$  do  
    for  $j := i + 1$  to  $n$  do  
      for  $k := i + j - 1$  to  $n$  do  
         $r := r + 1$   
      return( $r$ )
```

- 2-5. [5] Suppose the following algorithm is used to evaluate the polynomial

```
 $p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$   
  
 $p := a_0$ ;  
 $xpower := 1$ ;  
for  $i := 1$  to  $n$  do  
   $xpower := x * xpower$ ;  
   $p := p + a_i * xpower$   
end
```

- (a) How many multiplications are done in the worst-case? How many additions?
- (b) How many multiplications are done on the average?
- (c) Can you improve this algorithm?

- 2-6. [3] Prove that the following algorithm for computing the maximum value in an array $A[1..n]$ is correct.

```
function max( $A$ )  
   $m := A[1]$   
  for  $i := 2$  to  $n$  do  
    if  $A[i] > m$  then  $m := A[i]$   
  return ( $m$ )
```

Big Oh

- 2-7. [3] True or False?

- (a) Is $2^{n+1} = O(2^n)$?
- (b) Is $2^{2n} = O(2^n)$?

- 2-8. [3] For each of the following pairs of functions, either $f(n)$ is in $O(g(n))$, $f(n)$ is in $\Omega(g(n))$, or $f(n) = \Theta(g(n))$. Determine which relationship is correct and briefly explain why.

- (a) $f(n) = \log n^2$; $g(n) = \log n + 5$

- (b) $f(n) = \sqrt{n}$; $g(n) = \log n^2$
- (c) $f(n) = \log^2 n$; $g(n) = \log n$
- (d) $f(n) = n$; $g(n) = \log^2 n$
- (e) $f(n) = n \log n + n$; $g(n) = \log n$
- (f) $f(n) = 10$; $g(n) = \log 10$
- (g) $f(n) = 2^n$; $g(n) = 10n^2$
- (h) $f(n) = 2^n$; $g(n) = 3^n$

2-9. [3] For each of the following pairs of functions $f(n)$ and $g(n)$, determine whether $f(n) = O(g(n))$, $g(n) = O(f(n))$, or both.

- (a) $f(n) = (n^2 - n)/2$, $g(n) = 6n$
- (b) $f(n) = n + 2\sqrt{n}$, $g(n) = n^2$
- (c) $f(n) = n \log n$, $g(n) = n\sqrt{n}/2$
- (d) $f(n) = n + \log n$, $g(n) = \sqrt{n}$
- (e) $f(n) = 2(\log n)^2$, $g(n) = \log n + 1$
- (f) $f(n) = 4n \log n + n$, $g(n) = (n^2 - n)/2$

2-10. [3] Prove that $n^3 - 3n^2 - n + 1 = \Theta(n^3)$.

2-11. [3] Prove that $n^2 = O(2^n)$.

2-12. [3] For each of the following pairs of functions $f(n)$ and $g(n)$, give an appropriate positive constant c such that $f(n) \leq c \cdot g(n)$ for all $n > 1$.

- (a) $f(n) = n^2 + n + 1$, $g(n) = 2n^3$
- (b) $f(n) = n\sqrt{n} + n^2$, $g(n) = n^2$
- (c) $f(n) = n^2 - n + 1$, $g(n) = n^2/2$

2-13. [3] Prove that if $f_1(n) = O(g_1(n))$ and $f_2(n) = O(g_2(n))$, then $f_1(n) + f_2(n) = O(g_1(n) + g_2(n))$.

2-14. [3] Prove that if $f_1(n) = \Omega(g_1(n))$ and $f_2(n) = \Omega(g_2(n))$, then $f_1(n) + f_2(n) = \Omega(g_1(n) + g_2(n))$.

2-15. [3] Prove that if $f_1(n) = O(g_1(n))$ and $f_2(n) = O(g_2(n))$, then $f_1(n) \cdot f_2(n) = O(g_1(n) \cdot g_2(n))$.

2-16. [5] Prove for all $k \geq 1$ and all sets of constants $\{a_k, a_{k-1}, \dots, a_1, a_0\} \in R$,

$$a_k n^k + a_{k-1} n^{k-1} + \dots + a_1 n + a_0 = O(n^k)$$

2-17. [5] Show that for any real constants a and b , $b > 0$

$$(n + a)^b = \Theta(n^b)$$

2-18. [5] List the functions below from the lowest to the highest order. If any two or more are of the same order, indicate which.

n	2^n	$n \lg n$	$\ln n$
$n - n^3 + 7n^5$	$\lg n$	\sqrt{n}	e^n
$n^2 + \lg n$	n^2	2^{n-1}	$\lg \lg n$
n^3	$(\lg n)^2$	$n!$	$n^{1+\varepsilon}$ where $0 < \varepsilon < 1$

- 2-19. [5] List the functions below from the lowest to the highest order. If any two or more are of the same order, indicate which.

\sqrt{n}	n	2^n
$n \log n$	$n - n^3 + 7n^5$	$n^2 + \log n$
n^2	n^3	$\log n$
$n^{\frac{1}{3}} + \log n$	$(\log n)^2$	$n!$
$\ln n$	$\frac{n}{\log n}$	$\log \log n$
$(1/3)^n$	$(3/2)^n$	6

- 2-20. [5] Find two functions $f(n)$ and $g(n)$ that satisfy the following relationship. If no such f and g exist, write "None."

- (a) $f(n) = o(g(n))$ and $f(n) \neq \Theta(g(n))$
- (b) $f(n) = \Theta(g(n))$ and $f(n) = o(g(n))$
- (c) $f(n) = \Theta(g(n))$ and $f(n) \neq O(g(n))$
- (d) $f(n) = \Omega(g(n))$ and $f(n) \neq O(g(n))$

- 2-21. [5] True or False?

- (a) $2n^2 + 1 = O(n^2)$
- (b) $\sqrt{n} = O(\log n)$
- (c) $\log n = O(\sqrt{n})$
- (d) $n^2(1 + \sqrt{n}) = O(n^2 \log n)$
- (e) $3n^2 + \sqrt{n} = O(n^2)$
- (f) $\sqrt{n} \log n = O(n)$
- (g) $\log n = O(n^{-1/2})$

- 2-22. [5] For each of the following pairs of functions $f(n)$ and $g(n)$, state whether $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$, $f(n) = \Theta(g(n))$, or none of the above.

- (a) $f(n) = n^2 + 3n + 4$, $g(n) = 6n + 7$
- (b) $f(n) = n\sqrt{n}$, $g(n) = n^2 - n$
- (c) $f(n) = 2^n - n^2$, $g(n) = n^4 + n^2$

- 2-23. [3] For each of these questions, briefly explain your answer.

- (a) If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on some inputs?
- (b) If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on all inputs?
- (c) If I prove that an algorithm takes $\Theta(n^2)$ worst-case time, is it possible that it takes $O(n)$ on some inputs?

- (d) If I prove that an algorithm takes $\Theta(n^2)$ worst-case time, is it possible that it takes $O(n)$ on all inputs?
- (e) Is the function $f(n) = \Theta(n^2)$, where $f(n) = 100n^2$ for even n and $f(n) = 20n^2 - n \log_2 n$ for odd n ?
- 2-24. [3] For each of the following, answer *yes*, *no*, or *can't tell*. Explain your reasoning.
- (a) Is $3^n = O(2^n)$?
 - (b) Is $\log 3^n = O(\log 2^n)$?
 - (c) Is $3^n = \Omega(2^n)$?
 - (d) Is $\log 3^n = \Omega(\log 2^n)$?
- 2-25. [5] For each of the following expressions $f(n)$ find a simple $g(n)$ such that $f(n) = \Theta(g(n))$.
- (a) $f(n) = \sum_{i=1}^n \frac{1}{i}$.
 - (b) $f(n) = \sum_{i=1}^n \lceil \frac{1}{i} \rceil$.
 - (c) $f(n) = \sum_{i=1}^n \log i$.
 - (d) $f(n) = \log(n!)$.
- 2-26. [5] Place the following functions into increasing asymptotic order.
 $f_1(n) = n^2 \log_2 n$, $f_2(n) = n(\log_2 n)^2$, $f_3(n) = \sum_{i=0}^n 2^i$, $f_4(n) = \log_2(\sum_{i=0}^n 2^i)$.
- 2-27. [5] Place the following functions into increasing asymptotic order. If two or more of the functions are of the same asymptotic order then indicate this.
 $f_1(n) = \sum_{i=1}^n \sqrt{i}$, $f_2(n) = (\sqrt{n}) \log n$, $f_3(n) = n\sqrt{\log n}$, $f_4(n) = 12n^{\frac{3}{2}} + 4n$,
- 2-28. [5] For each of the following expressions $f(n)$ find a simple $g(n)$ such that $f(n) = \Theta(g(n))$. (You should be able to prove your result by exhibiting the relevant parameters, but this is not required for the homework.)
- (a) $f(n) = \sum_{i=1}^n 3i^4 + 2i^3 - 19i + 20$.
 - (b) $f(n) = \sum_{i=1}^n 3(4^i) + 2(3^i) - i^{19} + 20$.
 - (c) $f(n) = \sum_{i=1}^n 5^i + 3^{2i}$.
- 2-29. [5] Which of the following are true?
- (a) $\sum_{i=1}^n 3^i = \Theta(3^{n-1})$.
 - (b) $\sum_{i=1}^n 3^i = \Theta(3^n)$.
 - (c) $\sum_{i=1}^n 3^i = \Theta(3^{n+1})$.
- 2-30. [5] For each of the following functions f find a simple function g such that $f(n) = \Theta(g(n))$.
- (a) $f_1(n) = (1000)2^n + 4^n$.
 - (b) $f_2(n) = n + n \log n + \sqrt{n}$.
 - (c) $f_3(n) = \log(n^{20}) + (\log n)^{10}$.
 - (d) $f_4(n) = (0.99)^n + n^{100}$.

- 2-31. [5] For each pair of expressions (A, B) below, indicate whether A is O , o , Ω , ω , or Θ of B . Note that zero, one or more of these relations may hold for a given pair; list all correct ones.

	A	B
(a)	n^{100}	2^n
(b)	$(\lg n)^{12}$	\sqrt{n}
(c)	\sqrt{n}	$n^{\cos(\pi n/8)}$
(d)	10^n	100^n
(e)	$n^{\lg n}$	$(\lg n)^n$
(f)	$\lg(n!)$	$n \lg n$

Summations

- 2-32. [5] Prove that:

$$1^2 - 2^2 + 3^2 - 4^2 + \dots + (-1)^{k-1} k^2 = (-1)^{k-1} k(k+1)/2$$

- 2-33. [5] Find an expression for the sum of the i th row of the following triangle, and prove its correctness. Each entry is the sum of the three entries directly above it. All non existing entries are considered 0.

				1					
				1	1	1			
			1	2	3	2	1		
		1	3	6	7	6	3	1	
	1	4	10	16	19	16	10	4	1

- 2-34. [3] Assume that Christmas has n days. Exactly how many presents did my “true love” send me? (Do some research if you do not understand this question.)
- 2-35. [5] Consider the following code fragment.

```

for i=1 to n do
    for j=i to 2*i do
        output ‘‘foobar’’

```

Let $T(n)$ denote the number of times ‘foobar’ is printed as a function of n .

- Express $T(n)$ as a summation (actually two nested summations).
 - Simplify the summation. Show your work.
- 2-36. [5] Consider the following code fragment.

```

for i=1 to n/2 do
    for j=i to n-i do
        for k=1 to j do
            output ‘‘foobar’’

```

Assume n is even. Let $T(n)$ denote the number of times ‘foobar’ is printed as a function of n .

- Express $T(n)$ as three nested summations.
- Simplify the summation. Show your work.

- 2-37. [6] When you first learned to multiply numbers, you were told that $x \times y$ means adding x a total of y times, so $5 \times 4 = 5 + 5 + 5 + 5 = 20$. What is the time complexity of multiplying two n -digit numbers in base b (people work in base 10, of course, while computers work in base 2) using the repeated addition method, as a function of n and b . Assume that single-digit by single-digit addition or multiplication takes $O(1)$ time. (Hint: how big can y be as a function of n and b ?)
- 2-38. [6] In grade school, you learned to multiply long numbers on a digit-by-digit basis, so that $127 \times 211 = 127 \times 1 + 127 \times 10 + 127 \times 200 = 26,397$. Analyze the time complexity of multiplying two n -digit numbers with this method as a function of n (assume constant base size). Assume that single-digit by single-digit addition or multiplication takes $O(1)$ time.

Logarithms

- 2-39. [5] Prove the following identities on logarithms:
- (a) Prove that $\log_a(xy) = \log_a x + \log_a y$
 - (b) Prove that $\log_a x^y = y \log_a x$
 - (c) Prove that $\log_a x = \frac{\log_b x}{\log_b a}$
 - (d) Prove that $x^{\log_b y} = y^{\log_b x}$
- 2-40. [3] Show that $\lceil \lg(n+1) \rceil = \lfloor \lg n \rfloor + 1$
- 2-41. [3] Prove that the binary representation of $n \geq 1$ has $\lfloor \lg_2 n \rfloor + 1$ bits.
- 2-42. [5] In one of my research papers I give a comparison-based sorting algorithm that runs in $O(n \log(\sqrt{n}))$. Given the existence of an $\Omega(n \log n)$ lower bound for sorting, how can this be possible?

Interview Problems

- 2-43. [5] You are given a set S of n numbers. You must pick a subset S' of k numbers from S such that the probability of each element of S occurring in S' is equal (i.e., each is selected with probability k/n). You may make only one pass over the numbers. What if n is unknown?
- 2-44. [5] We have 1,000 data items to store on 1,000 nodes. Each node can store copies of exactly three different items. Propose a replication scheme to minimize data loss as nodes fail. What is the expected number of data entries that get lost when three random nodes fail?
- 2-45. [5] Consider the following algorithm to find the minimum element in an array of numbers $A[0, \dots, n]$. One extra variable tmp is allocated to hold the current minimum value. Start from $A[0]$; " tmp " is compared against $A[1]$, $A[2]$, \dots , $A[N]$ in order. When $A[i] < tmp$, $tmp = A[i]$. What is the expected number of times that the assignment operation $tmp = A[i]$ is performed?
- 2-46. [5] You have a 100-story building and a couple of marbles. You must identify the lowest floor for which a marble will break if you drop it from this floor. How fast can you find this floor if you are given an infinite supply of marbles? What if you have only two marbles?