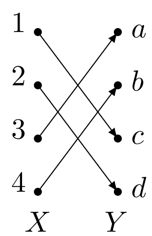
# Chapter 3

# Solutions to Selected Exercises

## Section 3.1

2. Not a function

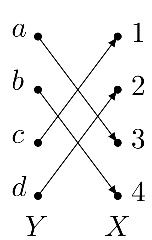
3. It is a function from $X$ to $Y$; domain $= X$, range $= Y$; it is both one-to-one and onto. Its arrow diagram is



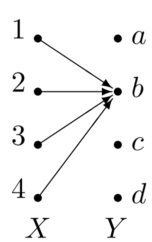

The inverse function is

$$\{(c,1),(d,2),(a,3),(b,4)\}.$$

For the inverse function, domain $= Y$, range $= X$. Its arrow diagram is

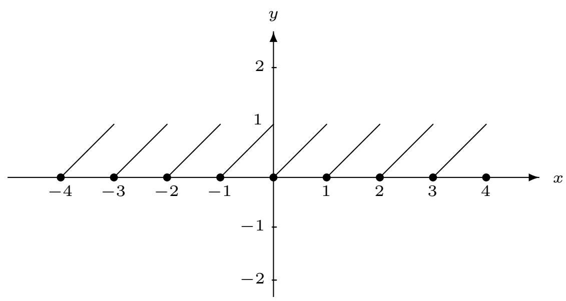



5. It is a function from $X$ to $Y$; domain $= X$, range $= \{b\}$. Its arrow diagram is

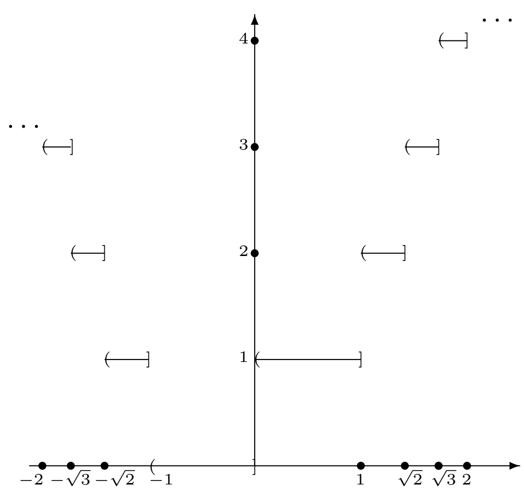



It is neither one-to-one nor onto.

7.




8.




11. Since $f(1) = f(-1)$, $f$ is not one-to-one. Since

$$f(n) = n^2 - 1 \geq -1,$$

for no $n$ do we have $f(n) = -2$. Therefore $f$ is not onto.

12. Since $f(1) = f(2)$, $f$ is not one-to-one. $f$ is onto. To prove this, let $m \in \mathbf{Z}$. Taking $n = 2m$, we have

$$f(n) = f(2m) = \lceil (2m)/2 \rceil = \lceil m \rceil = m.$$

Therefore $f$ is onto.

14. $f$ is one-to-one. To prove this, suppose that $f(m) = f(n)$. Then $2m = 2n$. Canceling 2, we have $m = n$. Therefore $f$ is one-to-one.

Since $f(n) = 2n$ is an even integer for every $n$, $f(n) \neq 3$ for every $n$. Therefore $f$ is not onto.

15. $f$ is one-to-one. To prove this, suppose that $f(m) = f(n)$. Then $m^3 = n^3$. Therefore $m = n$ and $f$ is one-to-one.

Since $f(n) = n^3$ is a cube for every $n$ and 5 is not a cube, $f(n) \neq 5$ for every $n$. Therefore $f$ is not onto.

17. Since $f(1, 0) = 1 = f(1, 1)$, $f$ is not one-to-one. $f$ is onto. To prove this, let $m \in \mathbf{Z}$. Then $f(m, 0) = m$. Therefore $f$ is onto.

18. Since $f(1,2) = 2 = f(2,1)$, $f$ is not one-to-one. $f$ is onto. To prove this, let $m \in \mathbf{Z}$. Then $f(m,1) = m$. Therefore $f$ is onto.

20. Since $f(0,1) = f(0,-1)$, $f$ is not one-to-one. Since
$$f(m,n) = n^2 + 1 \geq 1$$
for all $m$ and $n$, $f(m,n) \neq 0$ for all $m$ and $n$. Therefore $f$ is not onto.

21. Since $f(1,0) = f(0,1)$, $f$ is not one-to-one. $f$ is onto. To prove this, let $k \in \mathbf{Z}$. Then
$$f(k-2,0) = (k-2) + 0 + 2 = k.$$
Therefore $f$ is onto.

24. Not one-to-one. $f(4/3) = f(-2/3)$. Not onto. $f(x) \neq 0$ for any real $x$.

25. Not one-to-one. $\sin 0 = \sin 2\pi$. Not onto. $\sin x \neq 2$ for any real $x$.

27. One-to-one. Not onto. $f(x) \neq -2$ for any real $x$.

28. Not one-to-one. Notice that $f(x) = f(1/x)$. Thus any value of $x$, $x \neq 0$, $x \neq 1$, shows that $f$ is not one-to-one. Not onto. $f(x) \neq 1$ for any real $x$. (In fact, $-1/2 \leq f(x) \leq 1/2$ for all real $x$.)

30. Let $f$ be the function from $X = \{a,b\}$ to $Y = \{y\}$ given by $f = \{(a,y),(b,y)\}$.

31. The function $\{(1,1),(2,1)\}$ from $\{1,2\}$ to $\{1,2\}$.

33. $f^{-1}(y) = \log_3 y$ 34. $f^{-1}(y) = 2^{y/3}$ 36. $f^{-1}(y) = \left(\dfrac{y+5}{4}\right)^{1/3}$

37. $f^{-1}(y) = [\log_2(y-6)+1]/7$

39. $(f \circ f)(x) = 2(2n+1)+1$, $(g \circ g)(x) = 3(3n-1)-1$, $(f \circ g)(x) = 2(3n-1)+1$, $(g \circ f)(x) = 3(2n+1)-1$

40. $(f \circ f)(x) = n^4$, $(g \circ g)(x) = 2^{2^n}$, $(f \circ g)(x) = 2^{2n}$, $(g \circ f)(x) = 2^{n^2}$

43. $g(x) = 1/x$, $h(x) = 2x$, $w(x) = x^2$, $(g \circ h \circ w)(x) = f(x)$

44. $g(x) = 2x$, $h(x) = \sin x$, $f(x) = (h \circ g)(x)$

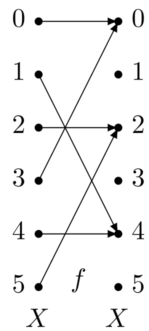
46. $g(x) = x^4$, $h(x) = 3 + x$, $w(x) = \sin x$, $(g \circ h \circ w)(x) = f(x)$

47. $g(x) = 1/x^3$, $h(x) = 6x$, $t(x) = \cos x$, $f(x) = (g \circ t \circ h)(x)$

49. 4; one-to-one functions: $\{(1,a),(2,b)\}$ and $\{(1,b),(2,a)\}$. In this case, the onto and one-to-one functions are the same.

50. (a) $f \circ f = \{(a,a),(b,b),(c,a)\}$, $f \circ f \circ f = \{(a,b),(b,a),(c,b)\}$
    (b) $f^9 = f$, $f^{623} = f$

52. $f = \{(0,0),(1,4),(2,2),(3,0),(4,4),(5,2)\}$. $f$ is neither one-to-one nor onto. The arrow diagram of $f$ is




56. $714 : 0, 631 : 2, 26 : 9, 373 : 16, 775 : 10, 906 : 5, 509 : 1, 2032 : 11, 42 : 8, 4 : 4, 136 : 3, 1028 : 12$

57. $53 : 4, 13 : 5, 281 : 3, 743 : 6, 377 : 9, 20 : 7, 10 : 1, 796 : 8$

59. During a search if we stop the search at an empty cell, we may not find the item even if it is present. The cell may be empty because an item was deleted. One solution is to mark deleted cells and consider them nonempty during a search.

60. No. If the data item is present, it will be found before an empty cell is encountered.

63. False. Let $X = \{1\}$, $Y = \{a, b\}$, $Z = \{\alpha, \beta\}$. A counterexample is $f = \{(a, \alpha), (b, \beta)\}$, $g = \{(1, a)\}$.

64. False. Let $X = \{1, 2\}$, $Y = \{a, b\}$, $Z = \{\alpha, \beta\}$. A counterexample is $f = \{(a, \alpha), (b, \alpha)\}$, $g = \{(1, a), (2, b)\}$.

66. True

67. False. Let $X = \{1\}$, $Y = \{a, b\}$, $Z = \{\alpha\}$. A counterexample is $f = \{(a, \alpha), (b, \alpha)\}$, $g = \{(1, a)\}$.

69. True. Let $z \in Z$. Since $f \circ g$ is onto, there exists $x \in X$ such that $f(g(x)) = z$. Let $y = g(x)$. Then $f(y) = z$. Therefore $f$ is onto $Z$.

71. Suppose that $f$ is not one-to-one. Then, for some $x$ and $y$, $f(x) = f(y)$, but $x \neq y$. Let $A = \{x\}$, $B = \{y\}$.

    Suppose that $f$ is one-to-one. Let $y \in f(A \cap B)$. Then $y = f(x)$ for some $x \in A \cap B$. Thus $y \in f(A) \cap f(B)$. Let $y \in f(A) \cap f(B)$. Then $y = f(a) = f(b)$, for some $a \in A$, $b \in B$. Since $f$ is one-to-one, $a = b$. Therefore, $y \in f(A \cap B)$.

72. [The case: If $g$ is one-to-one, then $f \circ g$ is one-to-one implies that $f$ is one-to-one.]

    Suppose that $f$ is not one-to-one. Then there exist distinct $x_1, x_2 \in X$ with $f(x_1) = f(x_2)$. Let $A = \{1, 2\}$, and let $g = \{(1, x_1), (2, x_2)\}$. Now $g$ is one-to-one, but $f \circ g$ is not one-to-one, which is a contradiction.

73. Suppose that $f$ is onto $Y$. Let $g$ be a function from $Y$ onto $Z$. We must show that $g \circ f$ is onto $Z$. Let $z \in Z$. Since $g$ is onto $Z$, there exists $y \in Y$, with $g(y) = z$. Since $f$ is onto $Y$, there exists $x \in X$, with $f(x) = y$. Now $g \circ f(x) = z$. Therefore, $g \circ f$ is onto.

    Suppose that whenever $g$ is a function from $Y$ onto $Z$, $g \circ f$ is onto $Z$. Suppose that $f$ is not onto $Y$. Then there exists $y_0 \in Y$ such that for no $x \in X$ do we have $f(x) = y_0$. Let $Z = \{0, 1\}$. Define $g$ from $Y$ to $Z$ by $g(y_0) = 1$, and $g(y) = 0$ if $y \neq y_0$. Then $g$ is onto $Z$, but $g \circ f$ is not onto $Z$.

74. If $x \in X$, then $x \in f^{-1}(f(\{x\}))$. Thus $\cup\{S \mid S \in \mathcal{S}\} = X$.

    Suppose that
    $$a \in f^{-1}(\{y\}) \cap f^{-1}(\{z\})$$
    for some $y, z \in Y$. Then $f(a) = y$ and $f(a) = z$. Thus $y = z$. Therefore, $\mathcal{S}$ is a partition of $X$.

76. $E_1$ is onto. To prove this, let $y \in \mathbf{R}$. Define $f\colon \mathbf{R} \to \mathbf{R}$ by the rule $f(x) = x - 1 + y$. Now
    $$E_1(f) = f(1) = 1 - 1 + y = y.$$
    Therefore $f$ is onto.

78. $S(\emptyset) = 000$

79. $S(X) = 111$

81. Let $s_1 s_2 s_3$ be a bit string of length 3. Define $Y$ as follows: Put $a$ in $Y$ if and only if $s_1 = 1$; put $b$ in $Y$ if and only if $s_2 = 1$; and put $c$ in $Y$ if and only if $s_3 = 1$. Then $Y \subseteq X$ and $S(Y) = s_1 s_2 s_3$. Therefore $S$ is onto.

83. If $x \in X - Y$, then
    $$C_{X \cup Y}(x) = 1 = 1 + 0 - 1 \cdot 0 = C_X(x) + C_Y(x) - C_X(x)C_Y(x).$$
    Similarly, if $x \in Y - X$, the equation holds. If $x \in X \cap Y$, then
    $$C_{X \cup Y}(x) = 1 = 1 + 1 - 1 \cdot 1 = C_X(x) + C_Y(x) - C_X(x)C_Y(x).$$
    If $x \notin X \cup Y$, then
    $$C_{X \cup Y}(x) = 0 = 0 + 0 - 0 \cdot 0 = C_X(x) + C_Y(x) - C_X(x)C_Y(x).$$
    Thus the equation holds for all $x \in U$.

84. If $x \in X$, then $x \notin \overline{X}$; thus,
    $$C_{\overline{X}}(x) = 0 = 1 - 1 = 1 - C_X(x).$$
    If $x \notin X$, then $x \in \overline{X}$; thus,
    $$C_{\overline{X}}(x) = 1 = 1 - 0 = 1 - C_X(x).$$

86. If $C_X(x) = 0$, the inequality obviously holds. If $C_X(x) = 1$, then $x \in X$. Since $X \subseteq Y$, $x \in Y$. Thus $C_Y(x) = 1$ also. Again, the inequality holds.

87. $C_{X\Delta Y}(x) = C_X(x) + C_Y(x) - 2C_X(x)C_Y(x)$

89. Suppose that there is a one-to-one function $f$ from $X$ to $Y$. Let $R$ be the range of $f$ and choose $a \in X$. If $y \in R$, let $g(y) = f^{-1}(y)$. If $y \in Y - R$, let $g(y) = a$. Then $g$ is a function from $Y$ onto $X$.

    Suppose that there is a function $g$ from $Y$ onto $X$. For each $x \in X$, choose one $y \in Y$ with $g(y) = x$. Define $f(x) = y$. Then $f$ is a one-to-one function from $X$ to $Y$.

91. $f$ is not a binary operator since the range of $f$ is not contained in $X$.

92. $f$ is a commutative, binary operator.

94. $f$ is a commutative, binary operator. To see this, note that if $x, y \in X$, then $1 \le xy$. Now
$$0 \le (x-y)^2 = x^2 - 2xy + y^2;$$
hence,
$$1 \le xy \le x^2 - xy + y^2 = f(x, y).$$

96. $f(X) = X \cup \{1\}$

97. Let $R$ denote the set
$$\{(y, x) \mid (x, y) \in f\}.$$
The set of $y$ such that $(y, x) \in R$ is $Y$ since $f$ is onto. If $(y, x), (y, x') \in R$, then $x = x'$ since $f$ is one-to-one. Thus $R$ is a function from $Y$ to $X$.

    For each $x \in X$, there is exactly one $y \in Y$ with $R(y) = x$ since $f$ is a function. Therefore $R$ is one-to-one and onto.

99. False. A counterexample is $x = y = 1.5$.

100. False. A counterexample is $x = 2,\ y = 2.6$.

102. Since $n$ is an odd integer, $n = 2k + 1$ for some integer $k$. Now
$$\left\lceil \frac{n^2}{4} \right\rceil = \left\lceil \frac{4k^2 + 4k + 1}{4} \right\rceil = \left\lceil k^2 + k + \frac{1}{4} \right\rceil = k^2 + k + 1$$
and
$$\frac{n^2 + 3}{4} = \frac{(4k^2 + 4k + 1) + 3}{4} = k^2 + k + 1.$$

103. $x = 1.5$

105. Suppose that $\lceil x \rceil = n$. By definition, $n$ is the least integer greater than or equal to $x$. Thus $x \le n$ and $n - 1$ is less than $x$, that is, $n - 1 < x \le n$. Thus, if we let $\varepsilon = n - x$, then $0 \le \varepsilon < 1$ and $x + \varepsilon = n$.

    Now suppose that there exists $\varepsilon$ satisfying $0 \le \varepsilon < 1$ and $x + \varepsilon = n$. Then $n - 1 < x \le n$ and so $n$ is the least integer greater than or equal to $x$. Therefore $\lceil x \rceil = n$.

106. For all real numbers $x$ and integers $n$, $\lfloor x \rfloor = n$ if and only if there exists $\varepsilon$, $0 \le \varepsilon < 1$, such that $x - \varepsilon = n$. The proof is similar to the proof in the solution to Exercise 105.

109. January, April, July

## Section 3.2

4. 5 5. 13 6. 199 7. 4153 8. 9 9. 45 10. 15

11. 3465 12. $s_n = 2(n+1) - 1$ 13. Yes 14. No 15. No

16. Yes 17. 8 18. 26 19. 41 20. 8 21. Yes 22. No

23. No 24. Yes 31. No 32. No 33. Yes 34. Yes

35. Yes 36. Yes 37. Yes 38. Yes 51. 2 52. 5

53. $c_n = n/2$, if $n$ is even; $c_n = (n-1)/2 - n$, if $n$ is odd.

54. $d_n = (-1)^{n/2} n!$, if $n$ is even; $d_n = (-1)^{(n+1)/2} n!$, if $n$ is odd.

55. No 56. No 57. No 58. No 59. 9 60. 30

61. $3n$ 62. $3^n$ 63. No 64. No 65. Yes 66. Yes

74. 3/4 75. 10/11 76. $1 - 1/(n+1)$ 77. $1/[(n+1)(n!)^2]$ 78. No

79. Yes 80. Yes 81. No 82. $3^n n!$ 87. $2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7$

88. $2^1, 2^2, 2^4, 2^7, 2^{11}, 2^{16}, 2^{22}$ 89. $n_k = \dfrac{k(k-1)+2}{2}$ 90. $t_{n_k} = 2^{n_k} = 2^{[k(k-1)+2]/2}$

95. $-1$ 96. $-14$ 97. $-88$ 98. $-476$ 99. $3 \cdot 2^p - 4 \cdot 5^p$

100. $3 \cdot 2^{n-1} - 4 \cdot 5^{n-1}$ 101. $3 \cdot 2^{n-2} - 4 \cdot 5^{n-2}$

102. $$\begin{aligned} 7r_{n-1} - 10r_{n-2} &\\ &= 7(3 \cdot 2^{n-1} - 4 \cdot 5^{n-1}) - 10(3 \cdot 2^{n-2} - 4 \cdot 5^{n-2}) \\ &= 3(7 \cdot 2^{n-1} - 10 \cdot 2^{n-2}) - 4(7 \cdot 5^{n-1} - 10 \cdot 5^{n-2}) \\ &= 3\left(\tfrac{7}{2}2^n - \tfrac{10}{4}2^n\right) - 4\left(\tfrac{7}{5}5^n - \tfrac{10}{25}5^n\right) \\ &= 3 \cdot 2^n - 4 \cdot 5^n = r_n \end{aligned}$$

103. 2 104. 9 105. 36 106. 135 107. $(2+i)3^i$ 108. $(1+n)3^{n-1}$

109. $n3^{n-2}$

110. $$\begin{aligned} 6z_{n-1} - 9z_{n-2} &= 6(1+n)3^{n-1} - 9(n3^{n-2}) = 2(1+n)3^n - n3^n \\ &= 3^n[2(1+n) - n] = (2+n)3^n = z_n \end{aligned}$$

112. $\displaystyle\sum_{k=0}^{n-1} (k+1)^2 r^{n-k-1}$ 113. $\displaystyle\sum_{i=0}^{n-1} C_i C_{n-i-1}$

115. The first sum is the sum by rows of the matrix

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ & a_{22} & a_{23} & \dots & a_{2n} \\ & & & \vdots & \\ & & & & a_{nn} \end{pmatrix},$$

and the second sum is the sum by columns of the same array. Thus the two sums are equal.

116. (a) *baabcaaba* (b) *caababaab* (c) *baabbaab* (d) *caabacaaba*
(e) 9 (f) 9 (g) 8 (h) 10 (i) *baab* (j) *caaba*
(k) *baabcaababbab* (l) *caabacaababbabbaab*

118. $\lambda, 0, 1, 00, 01, 10, 11$

119. $000, 001, 010, 011, 100, 101, 110, 111$

121. $\lambda, b, a, c, ba, ab, bc, bab, abc, babc$

122. $\lambda, a, b, aa, ab, ba, bb, aab, aba, baa, abb, aaba, abaa, baab, aabb,$
$aabaa, abaab, baabb, aabaab, abaabb, aabaabb$

124. The Basis Steps ($n = 1, 2$) are omitted.

For the Inductive Step, assume that $n \geq 3$ and the formula is true for $1 \leq i < n$. Then

$$\begin{aligned}
\frac{a_n}{n!} &= \frac{n-1}{n!}(a_{n-1} + a_{n-2}) \\
&= \frac{n-1}{n!}\left[(n-1)!\sum_{k=0}^{n-1}\frac{(-1)^k}{k!} + (n-2)!\sum_{k=0}^{n-2}\frac{(-1)^k}{k!}\right] \\
&= \frac{n-1}{n}\left[\sum_{k=0}^{n}\frac{(-1)^k}{k!} - \frac{(-1)^n}{n!}\right] + \frac{1}{n}\left[\sum_{k=0}^{n}\frac{(-1)^k}{k!} - \frac{(-1)^{n-1}}{(n-1)!} - \frac{(-1)^n}{n!}\right] \\
&= \sum_{k=0}^{n}\frac{(-1)^k}{k!}.
\end{aligned}$$

126. We prove only the case $x < x_1$. The case $x > x_n$ is treated similarly.

Since $x < x_1$, $|x - x_1| = x_1 - x$ and $|x - x_n| = x_n - x$. Thus

$$\begin{aligned}
\sum_{i=1}^{n}|x - x_i| &= (x_1 - x) + \sum_{i=2}^{n-1}|x - x_i| + (x_n - x) \\
&> (x - x_1) + \sum_{i=2}^{n-1}|x - x_i| + (x_n - x) \\
&= \sum_{i=2}^{n-1}|x - x_i| + (x_n - x_1).
\end{aligned}$$

127. We use strong induction.

Basis Steps ($n = 1, 2$).

If $n = 1$,

$$\sum_{i=1}^{1} |x - x_i| = |x - x_1|,$$

which is a minimum when $x = x_1$, the median.

Now suppose that $n = 2$. If $x_1 \le x \le x_2$, then

$$\sum_{i=1}^{2} |x - x_i| = (x - x_1) + (x_2 - x) = x_2 - x_1.$$

However, if $x < x_1$,

$$\sum_{i=1}^{2} |x - x_i| = (x_1 - x) + (x_2 - x) > (x - x_1) + (x_2 - x) = x_2 - x_1,$$

and if $x > x_2$,

$$\sum_{i=1}^{2} |x - x_i| = (x - x_1) + (x - x_2) > (x - x_1) + (x_2 - x) = x_2 - x_1.$$

Thus the minimum error, $x_2 - x_1$, is achieved when $x$ is a median, a value between $x_1$ and $x_2$. The proof of the Basis Step is complete.

Inductive Step. Suppose that the minimum error for $k < n$ values, $n \ge 3$, is achieved when $x$ is a median. We prove that the minimum error for $n$ values is achieved when $x$ is a median.

By the inductive assumption, the minimum error, which we denote $ME$, for $x_2, x_3, \ldots, x_{n-1}$, is achieved by taking $x$ to be a median of $x_2, x_3, \ldots, x_{n-1}$. If $x_1 \le x \le x_n$, by Exercise 125, we have

$$\sum_{i=1}^{n} |x - x_i| = \sum_{i=2}^{n-1} |x - x_i| + (x_n - x_1) \ge ME + (x_n - x_1).$$

The minimum value, $ME + (x_n - x_1)$, can be achieved by taking $x$ to be a median of $x_2, x_3, \ldots, x_{n-1}$. Since a median of $x_2, x_3, \ldots, x_{n-1}$ is equal to a median of $x_1, x_2, \ldots, x_n$, for $x$ satisfying $x_1 \le x \le x_n$ the minimum of $\sum_{i=1}^{n} |x - x_i|$ is achieved by taking $x$ to be a median of $x_1, x_2, \ldots, x_n$.

If $x < x_1$ or $x > x_n$, by Exercise 126

$$\sum_{i=1}^{n} |x - x_i| > \sum_{i=2}^{n-1} |x - x_i| + (x_n - x_1) \ge ME + (x_n - x_1).$$

Therefore, if $x < x_1$ or $x > x_n$, the sum $\sum_{i=1}^{n} |x - x_i|$ is *not* a minimum.

Therefore, the minimum of $\sum_{i=1}^{n} |x - x_i|$ is obtained by taking $x$ to be a median of $x_1, x_2, \ldots, x_n$. The proof of the Inductive Step is complete.

130. $f$ is one-to-one. Suppose that $f(\alpha) = f(\beta)$. We then have $\alpha\alpha = \beta\beta$. Now the first half of the first string, namely $\alpha$, must equal the first half of the second string, namely $\beta$. Therefore $f$ is one-to-one.

Notice that $|f(\alpha)|$ is an even integer for all $\alpha \in X^*$. Therefore $f(\alpha) \neq aaa$ for all $\alpha \in X^*$. Thus $f$ is not onto.

131. $f$ is one-to-one. Suppose that $f(\alpha) = f(\beta)$. We then have $\alpha\alpha^R = \beta\beta^R$. Now the first half of the first string, namely $\alpha$, must equal the first half of the second string, namely $\beta$. Therefore $f$ is one-to-one.

Notice that $|f(\alpha)|$ is an even integer for all $\alpha \in X^*$. Therefore $f(\alpha) \neq aaa$ for all $\alpha \in X^*$. Thus $f$ is not onto.

133. Taking $\alpha = \lambda$, the first rule tells us that $a\alpha b = ab \in L$ and $b\alpha a = ba \in L$. Taking $\alpha = ba$ and $\beta = ab$, the second rule tells us that $\alpha\beta = baab \in L$. Finally, taking $\alpha = baab$ and $\beta = ab$, the second rule tells us that $\alpha\beta = baabab \in L$.

134. Exercise 135 shows that if $\alpha \in L$, $\alpha$ has equal numbers of $a$'s and $b$'s. Since $aab$ has more $a$'s than $b$'s, $aab$ is not in $L$.

136. We use strong induction on the length $n$ of $\alpha$ to show that if $\alpha$ has equal numbers of $a$'s and $b$'s, then $\alpha \in L$. The Basis Step is $n = 0$. In this case, $\alpha$ is the null string, and the null string is in $L$ by the definition of $L$.

Suppose that $\alpha$ has length $n > 0$, and $\alpha$ has equal numbers of $a$'s and $b$'s. Notice that, because of the first rule, the length of $\alpha$ is at least two. First suppose that $\alpha$ starts with $a$ and ends with $b$, that is, $\alpha = a\beta b$. Then $\beta$ has length less than $n$, and $\beta$ has equal numbers of $a$'s and $b$'s. By the inductive assumption, $\beta \in L$. By the first rule, $\alpha = a\beta b \in L$. Similarly, if $\alpha$ starts with $a$ and ends with $b$, then $\alpha \in L$.

Now suppose that $\alpha$ has equal numbers of $a$'s and $b$'s and $\alpha$ starts with $a$ and ends with $a$, that is, $\alpha = a\beta a$. We claim that some proper substring of $\alpha$ starting at the beginning contains equal numbers of $a$'s and $b$'s; that is, we claim that $\alpha = \gamma\delta$, where $\gamma$ and $\delta$ have equal numbers of $a$'s and $b$'s, and neither $\gamma$ nor $\delta$ is the null string. Assuming that this claim is true, by the inductive assumption $\gamma$ and $\delta$ are in $L$, and it follows from the second rule that $\alpha \in L$. The Inductive Step is complete.

To prove the claim, for each substring $\varepsilon$ of $\alpha$ starting at the beginning, consider

$$\text{val}(\varepsilon) = \text{number of } a\text{'s} - \text{number of } b\text{'s}.$$

Consider val($\varepsilon$) for substrings $\varepsilon$ of increasing length. For the first substring, $a$, $\text{val}(a) = 1$. For the next-to-last substring ($\alpha$ with the trailing $a$ omitted), we have $\text{val}(\varepsilon) = -1$. When the length of the substring increases by one, val($\varepsilon$) increases or decreases by one. Since val's first value is 1 and its last value is $-1$, for some substring $\beta$, $\text{val}(\beta) = 0$. Thus $\beta$ has equal numbers of $a$'s and $b$'s, and the claim is proved.

137. Let $\varepsilon > 0$. Since $L - \varepsilon$ is not an upper bound of $\{a_n \mid n = 1, 2, \ldots\}$, there exists $N$ such that $L - \varepsilon < a_N$. Since $\{a_n\}_{n=1}^{\infty}$ is a nondecreasing sequence, for all $n \geq N$ we have $L - \varepsilon < a_N \leq a_n$. Since $L$ is an upper bound of $\{a_n \mid n = 1, 2, \ldots\}$, $a_n \leq L$ for all $n$.

## Section 3.3

2. $\{(a,3),(b,1),(b,4),(c,1)\}$

3. {(Sally,Math), (Ruth,Physics), (Sam,Econ)}

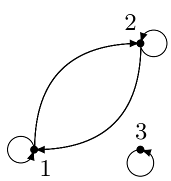
6.

| | |
|---|---|
| Roger | Music |
| Pat | History |
| Ben | Math |
| Pat | PolySci |

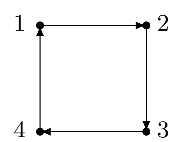
7.

| | |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 2 | 2 |
| 3 | 2 |
| 4 | 2 |
| 2 | 3 |
| 3 | 3 |
| 4 | 3 |
| 2 | 4 |
| 3 | 4 |
| 4 | 4 |

10.




11.




14. $\{(1,1),(2,2),(3,3),(3,5),(4,3),(4,4),(5,5),(5,4)\}$

15. $\emptyset$

19. $\{(1,1),(4,1),(2,2),(5,2),(3,3),(1,4),(4,4),(2,5),(5,5)\}$

21. $R = \{(1,2),(2,3),(3,4),(4,5)\}$
 $R^{-1} = \{(2,1),(3,2),(4,3),(5,4)\}$

22. Symmetric

25. Antisymmetric, transitive

26. Reflexive, antisymmetric, transitive, partial order

28. Reflexive, symmetric, transitive

29. Reflexive, symmetric, transitive

31. Symmetric

33. Let $R$ be a relation on a set $X$. Suppose that $R$ is antisymmetric. Let $(x,y) \in R$ with $x \neq y$. Suppose, by way of contradiction, $(y,x) \in R$. By the definition of antisymmetric, since $(x,y),(y,x) \in R$, $x = y$, which is a contradiction. Therefore $(y,x) \notin R$.

Now suppose that for all $x, y \in X$, if $(x, y) \in R$ and $x \neq y$, then $(y, x) \notin R$. We show that $R$ is antisymmetric. Let $(x, y), (y, x) \in R$. Suppose, by way of contradiction, that $x \neq y$. Since $(x, y) \in R$ and $x \neq y$, $(y, x) \notin R$, which is a contradiction. Therefore $x = y$ and so $R$ is antisymmetric.

34. Reflexive, symmetric

36. $R_1 \circ R_2 = \{(1,1), (1,2), (2,1), (2,2), (3,1), (3,2), (4,2)\}$,
$R_2 \circ R_1 = \{(1,1), (1,2), (3,4), (4,1), (4,2)\}$

38. $\{(1,1), (2,2), (3,3), (4,4), (1,2), (2,3)\}$

39. $\{(1,1), (2,2), (3,3), (4,4), (1,2), (2,3)\}$

41. $\{(1,2), (2,3), (1,3)\}$

43. True 44. True 46. True 47. True

49. False. Let $R = \{(2,3), (3,2)\}$, $S = \{(1,2), (2,1)\}$.

50. True 52. True

53. False. Let $R = \{(2,3), (1,1)\}$, $S = \{(1,2), (3,1)\}$.

55. Let $X$ be an $n$-element set. A relation on $X$ is a subset of $X \times X$. Since $|X \times X| = n^2$ and $|\mathcal{P}(X \times X)| = 2^{n^2}$, there are $2^{n^2}$ relations on an $n$-element set.

57. $R$ is reflexive, not symmetric, not antisymmetric, transitive, and not a partial order. To see that $R$ is not symmetric, consider $A = \{1\}$ and $B = \{1, 2\}$. To see that $R$ is not antisymmetric, consider $A =$ all real numbers and $B =$ all rational numbers.

58. $R$ is reflexive, symmetric, not antisymmetric, transitive, and not a partial order. To see that $R$ is not antisymmetric, consider $A =$ all real numbers and $B =$ all rational numbers.

59. If may be the case that for $x \in X$, there is no $y \in X$ such that $(x, y) \in R$. Consider, for example, $X = \{1, 2, 3\}$, $R = \{(1,1), (2,2), (1,2), (2,1)\}$, and $x = 3$.

## Section 3.4

2. Not an equivalence relation (not transitive)

3. Not an equivalence relation (not reflexive)

5. Equivalence relation. $[1] = [2] = [3] = [4] = [5] = \{1, 2, 3, 4, 5\}$.

6. Equivalence relation. $[1] = [5] = \{1, 5\}$, $[2] = \{2\}$, $[3] = \{3\}$, $[4] = \{4\}$.

8. Not an equivalence relation (not reflexive, not symmetric, not transitive)

10. Not an equivalence relation (not transitive) 11. Equivalence relation

13. Equivalence relation 14. Equivalence relation

16. $\{(1,1),(2,2),(3,3),(4,4),(3,4),(4,3)\}$, $[1]=\{1\}$, $[2]=\{2\}$, $[3]=[4]=\{3,4\}$

17. $\{(1,1),(2,2),(3,3),(4,4)\}$, $[i]=\{i\}$ for $i=1,\ldots,4$

19. $\{(i,j)\mid i,j\in\{1,2,3,4\}\}$, $[1]=[2]=[3]=[4]=\{1,2,3,4\}$

20. $\{(1,1),(2,2),(2,4),(4,2),(4,4),(3,3)\}$, $[1]=\{1\}$, $[2]=[4]=\{2,4\}$, $[3]=\{3\}$

21. Reflexive: $ARA$ since $A\cup Y=A\cup Y$.

    Symmetric: If $ARB$, then $A\cup Y=B\cup Y$. Now $B\cup Y=A\cup Y$, so $BRA$.

    Transitive: Suppose that $ARB$ and $BRC$. Then $A\cup Y=B\cup Y$ and $B\cup Y=C\cup Y$. Therefore $A\cup Y=C\cup Y$. Thus $ARC$.

23. Eight. An equivalence class is determined by the presence or absence of 1, 2, and 5.

25. If $R$ is a relation on $X$, then $R=X\times X$.

27. $\{(1,1),(2,2),(3,3),(4,4),(5,5),(6,6),(1,2),(2,1),(3,4),(4,3)\}$

28. Five, corresponding to the partitions $\{\{1\},\{2\},\{3\}\},\{\{1\},\{2,3\}\},\{\{1,2\},\{3\}\},\{\{1,3\},\{2\}\},$ $\{\{1,2,3\}\}$

30. Reflexive: For all $f\in\mathbf{R}^{\mathbf{R}}$, $f(0)=f(0)$. Therefore $fRf$ and $R$ is reflexive.

    Symmetric: Suppose that $fRg$. Then $f(0)=g(0)$. Since $g(0)=f(0)$, $gRf$ and $R$ is symmetric.

    Transitive: Suppose that $fRg$ and $gRh$. Then $f(0)=g(0)$ and $g(0)=h(0)$. Therefore $f(0)=g(0)$ and $fRh$. Thus $R$ is transitive.

    If $f(x)=x$ for all $x\in\mathbf{R}$, $[f]$ consists of all functions $g$ from $\mathbf{R}$ to $\mathbf{R}$ satisfying $g(0)=0$.

32. (a) Reflexive: $(a,b)R(a,b)$ for all $a,b\in X$ since $ab=ba$ for all $a,b\in X$.
    Symmetric: Suppose that $(a,b)R(c,d)$. Then $ad=bc$. Since $cb=da$, $(c,d)R(a,b)$.
    Transitive: Suppose that $(a,b)R(c,d)$ and $(c,d)R(e,f)$. Then $ad=bc$ and $cf=de$. Now $af=adf/d=bcf/d=bde/d=be$. Therefore $(a,b)R(e,f)$.

    (b) $(1,1),(1,2),(1,3),(1,4),(1,5),(1,6),(1,7),(1,8),(1,9),(1,10),(2,1),(2,3),(2,5),(2,7),$
    $(2,9),(3,1),(3,2),(3,4),(3,5),(3,7),(3,8),(3,10),(4,1),(4,3),(4,5),(4,7),(4,9),$
    $(5,1),(5,2),(5,3),(5,4),(5,6),(5,7),(5,8),(5,9),(6,1),(6,5),(6,7),(7,1),(7,2),$
    $(7,3),(7,4),(7,5),(7,6),(7,8),(7,9),(7,10),(8,1),(8,3),(8,5),(8,7),(8,9),(9,1),$
    $(9,2),(9,4),(9,5),(9,7),(9,8),(9,10),(10,1),(10,3),(10,7),(10,9)$

    (c) $(a,b)R(c,d)$ if and only if $\dfrac{a}{b}=\dfrac{c}{d}$.

33. We show symmetry only. Let $(a,b)\in R\cap R^{-1}$. Then $(a,b)\in R$, so $(b,a)\in R^{-1}$. Since $(a,b)\in R^{-1}$, $(b,a)\in R$. Thus $(b,a)\in R\cap R^{-1}$ and $R\cap R^{-1}$ is symmetric.

35. $R$ is reflexive since for every $x$, $x\in S$ for some $S\in\mathcal{S}$. $R$ is also symmetric, for suppose that $xRy$. Then $x,y\in S$ for some $S\in\mathcal{S}$. Thus $y,x\in S$ and $yRx$. $R$ need not be transitive. Let $X=\{1,2,3\}$, $S_1=\{1,2\}$, $S_2=\{2,3\}$. Then $X=S_1\cup S_2$. Now $1R2$ and $2R3$, but $1\not R3$.

36. (b) Cylinder

38. Reflexive: For every $x \in X$, by the definition of a function $f(x)$ is defined. Since $f(x) = f(x)$, $xRx$ for every $x \in X$.

    Symmetric: Suppose that $xRy$. Then $f(x) = f(y)$. Since $f(y) = f(x)$, $yRx$.

    Transitive: Suppose that $xRy$ and $yRz$. Then $f(x) = f(y)$ and $f(y) = f(z)$. Therefore $f(x) = f(z)$ and $xRz$.

39. Suppose that $f = C_Y$. The equivalence classes are $Y$ and $\overline{Y}$.

41. When $x$ and $y$ are in the same equivalence class

42. Suppose that $[x] = [y]$. Then $xRy$. Therefore, $g(x) = g(y)$.

44. Assume that the sets $[x]$ partition $X$. Since $R$ is not symmetric, there exists $(b, c) \in R$ and $(c, b) \notin R$. Therefore $c \notin [b]$. There exists $d$ such that $c \in [d]$. Therefore $(c, d) \in R$. Because $R$ is transitive, $(b, d) \in R$. Therefore $b \in [d]$. Since $b \in [c]$, $[c] = [d]$. There exists $f \in [b]$. Therefore $(f, b) \in R$. Because $R$ is transitive, $(f, c) \in R$. Therefore $f \in [c]$. It follows that $[b] = [c] = [d]$. Therefore $c \notin [b]$ and $c \in [d]$, which is a contradiction. Thus the collection of pseudo equivalence classes does not partition $X$.

45. Since $R$ is not symmetric, there exist $(a, b) \in R$ and $(b, a) \notin R$. Then $a \in [b]$, but $b \notin [a]$. Since $R$ is reflexive, $a \in [a]$ and $b \in [b]$. Therefore $[a] \cap [b] \neq \emptyset$ but $[a] \neq [b]$. Thus the collection of pseudo equivalence classes does not partition $X$.

47. Let $X = \{a, b, c\}$ and $R = \{(a, b), (b, c), (c, a)\}$. Then $[a] = \{c\}$, $[b] = \{a\}$, $[c] = \{b\}$ partitions $X$.

48. Let $X = \{a, b, c, d\}$ and

    $$R = \{(b, a), (b, c), (d, a), (d, c), (a, b), (a, d), (c, b), (c, d)\}.$$

    Then

    $$[a] = [c] = \{b, d\} \quad \text{and} \quad [b] = [d] = \{a, c\}$$

    partitions $X$.

49. (b) Sequences are equivalent if their domains are the same size and their first range values agree, their second range values agree, and so on.

    (c) For two sequences to be equal, their domains must be *equal* and their first range values agree, their second range values agree, and so on. Equivalent sequences can have different domains.

51. Since $(y, y) \in \{(x, x) \mid x \in X\}$ for all $y \in X$, $(y, y) \in \rho(R)$ for all $y \in X$. Thus $\rho(R)$ is reflexive.

52. Let $(x, y)$ be in $R \cup R^{-1}$. If $(x, y)$ is in $R$, $(y, x)$ is in $R^{-1}$, so $(y, x)$ is in $R \cup R^{-1}$. If $(x, y)$ is in $R^{-1}$, then $(y, x)$ is in $R$, so $(y, x)$ is in $R \cup R^{-1}$. In any case, if $(x, y)$ is in $R \cup R^{-1}$, $(y, x)$ is in $R \cup R^{-1}$, so $R \cup R^{-1}$ is symmetric.

54. Since

$$R \subseteq \rho(R), \quad R \subseteq \sigma(R), \quad R \subseteq \tau(R), \tag{3.1}$$

it follows that $R \subseteq \tau(\sigma(\rho(R)))$.

By (3.1), $\rho(R) \subseteq \tau(\sigma(\rho(R)))$ and by Exercise 51, $\rho(R)$ is reflexive. Therefore, $\tau(\sigma(\rho(R)))$ is reflexive.

By Exercise 52, $\sigma(\rho(R))$ is symmetric. We show that if $R'$ is any symmetric relation, $\tau(R')$ is symmetric. We can then conclude that $\tau(\sigma(\rho(R)))$ is symmetric.

Let $R'$ be a symmetric relation. Let $(x, y) \in \tau(R')$. Then there exist $x = x_0, \ldots, x_n = y \in X$ such that $(x_{i-1}, x_i) \in R'$ for $i = 1, \ldots, n$. Since $R'$ is symmetric, $(x_i, x_{i-1}) \in R'$ for $i = 1, \ldots, n$. Thus $(y, x) \in \tau(R')$ and $\tau(R')$ is symmetric.

By Exercise 53, $\tau(\sigma(\rho(R)))$ is transitive; hence $\tau(\sigma(\rho(R)))$ is an equivalence relation containing $R$.

55. By Exercise 54, $\tau(\sigma(\rho(R)))$ is an equivalence relation containing $R$.

We first observe that if $R_1 \subseteq R_2 \subseteq X \times X$, then

$$\rho(R_1) \subseteq \rho(R_2), \quad \sigma(R_1) \subseteq \sigma(R_2), \quad \tau(R_1) \subseteq \tau(R_2).$$

It follows that if $R'$ is a relation on $X$ and $R' \supseteq R$,

$$\tau(\sigma(\rho(R'))) \supseteq \tau(\sigma(\rho(R))).$$

The conclusion will follow if we show that if $R'$ is an equivalence relation on $X$, $R' = \tau(\sigma(\rho(R')))$.

Suppose that $R'$ is an equivalence relation. Since $R'$ is reflexive, $\rho(R') = R'$. Since $R'$ is symmetric, $\sigma(R') = R'$. Thus $\sigma(\rho(R')) = R'$. We show that $\tau(R') = R'$. Clearly, $R' \subseteq \tau(R')$. Let $(x, y) \in \tau(R')$. Then $(x, y) \in R'^n$ for some positive integer $n$. Thus there exist $x_0, \ldots, x_n \in X$ with $x = x_0$, $y = x_n$, and $(x_{i-1}, x_i) \in R'$ for $i = 1, \ldots, n$. Since $R$ is transitive, $(x, y) = (x_0, x_n) \in R'$. Thus $R' \supseteq \tau(R')$. Therefore, $R' = \tau(R')$. Now

$$R' = \tau(R') = \tau(\sigma(\rho(R'))).$$

58. False. Let $R_1 = \{(1, 1), (1, 2)\}$, $R_2 = \{(2, 2), (2, 1)\}$.

59. False. Let $X = \{1, 2, 3\}$, $R_1 = \{(1, 2)\}$, $R_2 = \{(2, 3)\}$.

61. False. Let $R_1 = \{(1, 2), (2, 3)\}$.

62. True

65. The function $f(n) = 2n$ is a one-to-one, onto function from $\{1, 2, \ldots\}$ to $\{2, 4, \ldots\}$.

66. Assume that $X$ is equivalent to $\mathcal{P}(X)$. Then there is a one-to-one, onto function $f$ from $X$ to $\mathcal{P}(X)$. Let

$$Y = \{x \in X \mid x \notin f(x)\}.$$

Then $f(y) = Y$ for some $y \in X$. Consider the possibilities $y \in Y$ and $y \notin Y$.

## Section 3.5

2. $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

3. $\begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

5. $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$

6. $\begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

7. [For Exercise 13] $\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

9. $\{(1,1),(1,3),(2,2),(2,3),(2,4)\}$

10. $\{(w,w),(w,y),(y,w),(y,y),(z,z)\}$

12. Symmetric and transitive

13. Take the transpose of the given matrix.

15. [For Exercise 4] The matrix of $R$ is

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

and its square is

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The entry in row 1, column 3 of the square is nonzero, but the entry in row 1, column 3 of the original matrix is zero. Therefore the relation is not transitive.

17. (a) $\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ (b) $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$ (c) $\begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

(d) $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

(e) $\{(2,1),(2,2),(2,3),(3,1),(3,2),(4,1),(4,2),(4,3),(5,1),(5,2),(5,3),(5,4)\}$

18. (a) $\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$ (b) $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$ (c) $\begin{pmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$

(d) The matrix of part (c) is the matrix of the relation $R_2 \circ R_1$.

(e) $\{(1,1),(1,2),(1,3),(1,4),(2,1),(2,2),(2,3),(3,1),(3,2),(4,1)\}$

20. Let $a_{ij}$ denote the $ij$th entry of $A_1$ and $b_{ij}$ denote the $ij$th entry of $A_2$. Let $n$ be the number of elements in $Y$. The $ik$th entry of $A_1A_2$ is found by taking the product of the $i$th row of $A_1$ and the $k$th column of $A_2$. Thus if $c_{ik}$ is the $ik$th entry of the product, we have

$$c_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk}.$$

Now a term in this sum is nonzero only when both factors in the term are nonzero. This will be the case for all $j$ such that $a_{ij} = b_{jk} = 1$. This happens only when $(i,j) \in R_1$ and $(j,k) \in R_2$. $c_{ik}$ is then precisely the number of these $j$'s.

22. Suppose that the $ij$th entry of $A$ is 1. Then the $ij$th entries of both $A_1$ and $A_2$ are 1. Thus $(i,j) \in R_1$ and $(i,j) \in R_2$. Therefore $(i,j) \in R_1 \cap R_2$. Now suppose that $(i,j) \in R_1 \cap R_2$. Then the $ij$th entries of both $A_1$ and $A_2$ are 1. Therefore the $ij$th entry of $A$ is 1. It follows that $A$ is the matrix of $R_1 \cap R_2$.

23. The matrix $A$ of Exercise 21 is

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix},$$

whose relation is

$$R_1 \cup R_2 = \{(1,1),(1,2),(2,2),(2,3),(3,1),(3,3)\}.$$

24. The matrix $A$ of Exercise 22 is

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix},$$

whose relation is

$$R_1 \cap R_2 = \{(2,2),(3,1),(3,3)\}.$$

26. Every column must contain at least one 1.

27. Every column must contain at most one 1.

## Section 3.6

2. {(23,Jones), (04,Yu), (96, Zamora), (66,Washington)}

3. {(04,335B2,220), (23,2A,14), (04,8C200,302), (66,42C,3), (04,900,7720), (96,20A8,200), (96,1199C,296), (23,772,39)}

4. {(United Supplies,2A), (ABC Unlimited,8C200), (United Supplies,1199C), (JCN Electronics,2A), (United Supplies,335B2), (ABC Unlimited,772), (Danny's,900), (United Supplies,772), (Underhanded Sales,20A8), (Danny's,20A8), (DePaul University,42C), (ABC Unlimited,20A8)}

6. DEPARTMENT[Manager]

   Jones, Yu, Zamora, Washington

7. SUPPLIER[Part No]

   335B2, 2A, 8C200, 42C, 900, 20A8, 1199C, 772

9. TEMP1 := EMPLOYEE[Manager = Jones]
   TEMP2 := TEMP1[Name]

   Kaminski, Schmidt, Manacotti

10. TEMP := SUPPLIER[Dept = 96]
    TEMP[Part No]

    20A8, 1199C

12. TEMP1 := DEPARTMENT[Dept = 04]
    TEMP2 := TEMP1[TEMP1.Manager = EMPLOYEE.Manager] EMPLOYEE
    TEMP3 := TEMP2[Name]

    Jones, Beaulieu

13. TEMP := SUPPLIER[Amount $\geq$ 100]
    TEMP[Part No]

    335B2, 8C200, 900, 20A8, 1199C

15. TEMP1 := BUYER[Name = United Supplies]
    TEMP2 := TEMP1[TEMP1.Part No = SUPPLIER.Part No] SUPPLIER
    TEMP3 := TEMP2[Part No, Amount]

| Part No | Amount |
|---|---|
| 335B2 | 220 |
| 2A | 14 |
| 1199C | 296 |
| 772 | 39 |

16. TEMP1 := BUYER[Name = ABC Unlimited]
    TEMP2 := TEMP1 [Part No = Part No] SUPPLIER
    TEMP3 := TEMP2 [Dept = Dept] DEPARTMENT
    TEMP3[Manager]

    Yu, Jones, Zamora

18. TEMP1 := DEPARTMENT[Manager = Jones]
    TEMP2 := TEMP1[TEMP1.Dept = SUPPLIER.Dept] SUPPLIER
    TEMP3 := TEMP2[TEMP2.Part No = BUYER.Part No] BUYER
    TEMP4 := TEMP3[Name]

    United Supplies, JCN Electronics ABC Unlimited

19. TEMP1 := EMPLOYEE[Name = Suzuki]
    TEMP2 := TEMP1 [Manager = Manager] DEPARTMENT
    TEMP3 := TEMP2 [Dept = Dept] SUPPLIER
    TEMP4 := TEMP3 [Part No = Part No] BUYER
    TEMP4[Name]

    Underhanded Sales, Danny's, ABC Unlimited, United Supplies

23. The intersection operator will operate on two relations with the same set of attributes (arranged in the same order). The relation resulting from the intersection will have the same set of attributes. A tuple in the new relation will be a tuple in both of the two relations operated on. We will express the intersection operation using the set intersection symbol.

    TEMP1 := BUYER[Part No = 2A]
    TEMP2 := TEMP1[Name]
    TEMP3 := BUYER[Part No = 1199C]
    TEMP4 := TEMP3[Name]
    TEMP5 := TEMP2 $\cap$ TEMP4

24. Let $R_1$ and $R_2$ be two $n$-ary relations. The *difference* of $R_1$ and $R_2$ is the $n$-ary relation $R_1 - R_2$.

    TEMP1 := EMPLOYEE [Manager = Manager] DEPARTMENT
    TEMP2 := TEMP1[Dept = 04]
    TEMP3 := TEMP1 $-$ TEMP2
    TEMP3[Name]

    Suzuki, Kaminski, Ryan, Schmidt, Manacotti

# Chapter 4

# Solutions to Selected Exercises

## Section 4.1

3.
```
min(a, b, c) {
    small = a
    if (b < small)
        small = b
    if (c < small)
        small = c
    return small
}
```

4.
```
second_smallest(a, b, c) {
    x = a
    y = b
    z = c
    if (x > y){
        temp = x
        x = y
        y = temp
    }
    if (y > z) {
        temp = y
        y = z
        z = temp
    }
    if (x > y) {
        temp = x
        x = y
        y = temp
    }
    return y
}
```

6.
```
find_large_2nd_large(s, n, large, second_largest) {
    if (s_1 < s_2) {
        large = s_2
        second_largest = s_1
    }
    else {
        large = s_1
        second_largest = s_2
    }
    for i = 3 to n
        if (s_i > second_largest)
            if (s_i > large) {
                second_largest = large
                large = s_i
            }
            else
                second_largest = s_i
}
```

7.
```
find_small_2nd_small(s, n, small, second_smallest) {
    if (s_1 < s_2) {
        small = s_1
        second_smallest = s_2
    }
    else {
        small = s_2
        second_smallest = s_1
    }
    for i = 3 to n
        if (s_i < second_smallest)
            if (s_i < small) {
                second_smallest = small
                small = s_i
            }
            else
                second_smallest = s_i
}
```

9.
```
find_largest_element(s, n) {
    large = s_1
    index_large = 1
    for i = 2 to n
        if (s_i > large) {
            large = s_i
            index_large = i
        }
```

```
    return index_large
}
```

10.
```
find_last_largest_element(s, n) {
    large = s_1
    index_large = 1
    for i = 2 to n
        if (s_i ≥ large) {
            large = s_i
            index_large = i
        }
    return index_large
}
```

12.
```
find_out_of_order1(s, n) {
    for i = 2 to n
        if (s_i < s_{i-1})
            return i
    return 0
}
```

13.
```
find_out_of_order2(s, n) {
    for i = 2 to n
        if (s_i > s_{i-1})
            return i
    return 0
}
```

15. Assume that $s_n, s_{n-1}, \ldots, s_1$ and $t_n, t_{n-1}, \ldots, t_1$ are the decimal representations of the two numbers to be added. The output is $u_{n+1}, u_n, \ldots, u_1$.

```
add(s, t, u, n) {
    c = 0
    for i = 1 to n {
        Let xy be the decimal representation of the sum c + s_i + t_i.
        u_i = y
        c = x
    }
    u_{n+1} = c
}
```

16.
```
transpose(A, n) {
    for i = 1 to n - 1
        for j = i + 1 to n
            swap(A_ij, A_ji)
}
```

18. The input is the $n \times n$ matrix $A$ of the relation, and $n$.

```
is_symmetric(A, n) {
   for i = 1 to n − 1
      for j = i + 1 to n
         if (A_ij ¬= A_ji)
            return false
   return true
}
```

19. The input is the $n \times n$ matrix $A$ of the relation, and $n$.

```
is_transitive(A, n) {
   // first compute B = A^2
   for i = 1 to n
      for j = 1 to n {
         B_ij = 0
         for k = 1 to n
            B_ij = B_ij + A_ik * A_kj
      }
   // if an entry in A^2 is nonzero, but the corresponding entry
   // in A is zero, the relation is not transitive
   for i = 1 to n
      for j = 1 to n {
         if (B_ij ¬= 0 ∧ A_ij == 0)
            return false
   return true
}
```

21. The input is $A$, the $m \times n$ matrix of the relation, and $m$ and $n$.

```
is_function(A, m, n) {
   for i = 1 to m {
      sum = 0
      for j = 1 to n
         sum = sum + A_ij
      if (sum ¬= 1)
         return false
   }
   return true
}
```

22. The input is the $n \times n$ matrix $A$ of the relation, and $n$.

```
inverse(A, n) {
   for i = 1 to n − 1
      for j = i + 1 to n
         swap(A_ij, A_ji)
}
```

24.
```
pair_sum(s, n, x) {
    for i = 1 to n − 1
        for j = i + 1 to n
            if (x == s_i + s_j)
                return true
    return false
}
```

## Section 4.2

2. First $i$ and $j$ are set to 1. The while loop then compares $t_1t_2t_3 =$ "bal" with $p =$ "lai". Since "b" and "l" are not equal, $i$ increments to 2 and $j$ remains 1.

   The while loop then compares $t_2t_3t_4 =$ "ala" with $p =$ "lai". Since "a" and "l" are not equal, $i$ increments to 3 and $j$ remains 1.

   The while loop then compares $t_3t_4t_5 =$ "lal" with $p =$ "lai". Since "l" and "l" are equal, $j$ increments. Since "a" and "a" are equal, $j$ increments again. Since "l" and "i" are not equal, $i$ increments to 4 and $j$ is reset to 1.

   The while loop then compares $t_4t_5t_6 =$ "ala" with $p =$ "lai". Since "a" and "l" are not equal, $i$ increments to 5 and $j$ remains 1.

   The while loop then compares $t_5t_6t_7 =$ "lai" with $p =$ "lai". Since the comparison succeeds, the algorithm returns $i = 5$ to indicate that $p$ was found in $t$ starting at index 5 in $t$.

3. First $i$ and $j$ are set to 1. The while loop then compares $t_1t_2t_3 =$ "000" with $p =$ "001". Since "0" and "0" are equal, $j$ increments. Since "0" and "0" are equal, $j$ increments again. Since "0" and "1" are not equal, $i$ increments to 2 and $j$ is reset to 1.

   The while loop then compares $t_2t_3t_4 =$ "000" with $p =$ "001". Since "0" and "0" are equal, $j$ increments. Since "0" and "0" are equal, $j$ increments again. Since "0" and "1" are not equal, $i$ increments to 3 and $j$ is reset to 1.

   This pattern repeats until the first for loop terminates. The algorithm then returns 0 to indicate the $p$ was not found in $t$.

5. First 20 is inserted in

   $$\boxed{34}$$

   Since $20 < 34$, 34 must move one position to the right

   $$\boxed{\phantom{34}}\boxed{34}$$

   Now 20 is inserted

   $$\boxed{20}\boxed{34}$$

Since $19 < 34$, 34 must move one position to the right

| | | |
|---|---|---|
| 20 | | 34 |

Since $19 < 20$, 20 must move one position to the right

| | | |
|---|---|---|
| | 20 | 34 |

Now 19 is inserted

| | | |
|---|---|---|
| 19 | 20 | 34 |

Since $5 < 34$, 34 must move one position to the right

| | | | |
|---|---|---|---|
| 19 | 20 | | 34 |

Since $5 < 20$, 20 must move one position to the right

| | | | |
|---|---|---|---|
| 19 | | 20 | 34 |

Since $5 < 19$, 19 must move one position to the right

| | | | |
|---|---|---|---|
| | 19 | 20 | 34 |

Now 5 is inserted

| | | | |
|---|---|---|---|
| 5 | 19 | 20 | 34 |

The sequence is now sorted.

6. Since $55 > 34$, it is immediately inserted to 34's right

| | |
|---|---|
| 34 | 55 |

Since $144 > 55$, it is immediately inserted to 55's right

| | | |
|---|---|---|
| 34 | 55 | 144 |

Since $259 > 144$, it is immediately inserted to 144's right

| | | | |
|---|---|---|---|
| 34 | 55 | 144 | 259 |

The sequence is now sorted.

9. We first swap $a_i$ and $a_j$, where $i = 1$ and $j = \mathit{rand}(1, 5) = 2$. After the swap we have

$$\begin{array}{ccccc} \hline \multicolumn{1}{|c|}{57} & \multicolumn{1}{c|}{34} & \multicolumn{1}{c|}{72} & \multicolumn{1}{c|}{101} & \multicolumn{1}{c|}{135} \\ \hline \uparrow & \uparrow & & & \\ i & j & & & \end{array}$$

We next swap $a_i$ and $a_j$, where $i = 2$ and $j = \mathit{rand}(2, 5) = 5$. After the swap we have

$$\begin{array}{ccccc} \hline \multicolumn{1}{|c|}{57} & \multicolumn{1}{c|}{135} & \multicolumn{1}{c|}{72} & \multicolumn{1}{c|}{101} & \multicolumn{1}{c|}{34} \\ \hline & \uparrow & & & \uparrow \\ & i & & & j \end{array}$$

We next swap $a_i$ and $a_j$, where $i = 3$ and $j = \mathit{rand}(3, 5) = 3$. The sequence is unchanged.

We next swap $a_i$ and $a_j$, where $i = 4$ and $j = \mathit{rand}(4, 5) = 4$. The sequence is again unchanged.

10. We first swap $a_i$ and $a_j$, where $i = 1$ and $j = \mathit{rand}(1, 5) = 5$. After the swap we have

$$\begin{array}{ccccc} \hline \multicolumn{1}{|c|}{135} & \multicolumn{1}{c|}{57} & \multicolumn{1}{c|}{72} & \multicolumn{1}{c|}{101} & \multicolumn{1}{c|}{34} \\ \hline \uparrow & & & & \uparrow \\ i & & & & j \end{array}$$

We next swap $a_i$ and $a_j$, where $i = 2$ and $j = \mathit{rand}(2, 5) = 5$. After the swap we have

$$\begin{array}{ccccc} \hline \multicolumn{1}{|c|}{135} & \multicolumn{1}{c|}{34} & \multicolumn{1}{c|}{72} & \multicolumn{1}{c|}{101} & \multicolumn{1}{c|}{57} \\ \hline & \uparrow & & & \uparrow \\ & i & & & j \end{array}$$

We next swap $a_i$ and $a_j$, where $i = 3$ and $j = \mathit{rand}(3, 5) = 4$. After the swap we have

$$\begin{array}{ccccc} \hline \multicolumn{1}{|c|}{135} & \multicolumn{1}{c|}{34} & \multicolumn{1}{c|}{101} & \multicolumn{1}{c|}{72} & \multicolumn{1}{c|}{57} \\ \hline & & \uparrow & \uparrow & \\ & & i & j & \end{array}$$

We next swap $a_i$ and $a_j$, where $i = 4$ and $j = \mathit{rand}(4, 5) = 4$. The sequence is unchanged.

12. Use the invariant: $s_1, \ldots, s_i$ is sorted.

13. *find_first_key*$(s, n, \mathit{key})$ {
    for $i = 1$ to $n$
        if $(\mathit{key} == s_i)$
            return $i$
    return 0
}

15. *insert*$(s, n, x)$ {
    $i = 1$
    while $(i \leq n \land x > s_i)$
        $i = i + 1$
    // move $s_i, \ldots, s_n$ down to make room for $x$
    $j = n$
    while $(j \geq i)$ {
        $s_{j+1} = s_j$
        $j = j - 1$
    }
    // insert $x$
    $s_i = x$
}

16. Replace the line

    while $(t_{i+j-1} == p_j)$ {

    by

    while $(j \leq m \land t_{i+j-1} == p_j)$ {

    Replace the line

    return $i$

    by

    *println*$(i)$

    and remove the line

    return 0

18. The worst case occurs when the for loop and the while loop run as long as possible. This situation is achieved when $t$ consists of $n$ 0's and $p$ consists of $m - 1$ 0's followed by one 1.

19. *insertion_sort_nonincreasing*$(s, n)$ {
    for $i = 2$ to $n$ {
        $val = s_i$ // save $s_i$ so it can be inserted into the correct place
        $j = i - 1$
        // if $val > s_j$, move $s_j$ right to make room for $s_i$
        while $(j \geq 1 \ \land val > s_j)$ {
            $s_{j+1} = s_j$
            $j = j - 1$
        }
        $s_{j+1} = val$ // insert $val$
    }
}

21. [For Exercise 4] Selection sort first finds the smallest item, $s_2$, and places it first by swapping $s_1$ and $s_2$. The result is

$$\begin{array}{|c|c|c|c|}\hline 20 & 34 & 144 & 55 \\ \hline \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\ \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \end{array}$$

Selection sort next finds the smallest item, $s_2$, in $s_2, s_3, s_4$, and places it second by swapping $s_2$ and $s_2$. The sequence is unchanged.

Selection sort next finds the smallest item, $s_4$, in $s_3, s_4$, and places it third by swapping $s_3$ and $s_4$. The result is

$$\begin{array}{|c|c|c|c|}\hline 20 & 34 & 55 & 144 \\ \hline \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{\uparrow} \\ \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{4} \end{array}$$

The sequence is now sorted.

22. In the pseudocode

```
selection_sort(s, n) {
   for i = 1 to n − 1 {
      // find smallest in s_i, ..., s_n
      small_index = i
      for j = i + 1 to n
         if (s_j < s_small_index)
            small_index = j
      swap(s_i, s_small_index)
   }
}
```

the for loops always run to completion regardless of the input.

## Section 4.3

2. $\Theta(n^2)$ 3. $\Theta(n^3)$ 5. $\Theta(n \lg n)$ 6. $\Theta(n^6)$ 8. $\Theta(n^2)$

9. $\Theta(n \lg n)$ 11. $\Theta(n)$ 12. $\Theta(2^n)$ 14. $\Theta(n^3)$ 15. $\Theta(n^{5/2})$

17. $\Theta(n)$ 18. $\Theta(n^2)$ 20. $\Theta(n^2)$ 21. $\Theta(n^3)$ 23. $\Theta(n^3)$

24. $\Theta(n)$ 26. $\Theta(\lg \lg n)$ 27. $\Theta(n)$

30. (a) Even: $3n/2 - 2$ Odd: $(3n - 1)/2 - 1$ (b) $\Theta(n)$

32. Use Example 4.3.6.

34. $2^n = \underbrace{2 \cdot 2 \cdots 2}_{n \text{ 2's}} = 2 \cdot \underbrace{2 \cdot 2 \cdots 2}_{n-1 \text{ 2's}} \le 2(2 \cdot 3 \cdots n) = 2n!$

35. First note that

$$\sum_{i=1}^{n} i \lg i \le n(n \lg n) = n^2 \lg n.$$

Therefore

$$\sum_{i=1}^{n} i \lg i = O(n^2 \lg n).$$

Now

$$\sum_{i=1}^{n} i \lg i \ge \sum_{i=\lceil n/2 \rceil}^{n} i \lg i \ge \sum_{i=\lceil n/2 \rceil}^{n} \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil = \left\lceil \frac{n+1}{2} \right\rceil \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil \ge \left( \frac{n}{2} \right)^2 \lg \left( \frac{n}{2} \right).$$

As in Example 4.3.9, if $n \ge 4$,

$$\left( \frac{n}{2} \right) \lg \left( \frac{n}{2} \right) \ge \frac{n \lg n}{4}.$$

Therefore if $n \ge 4$,

$$\sum_{i=1}^{n} i \lg i \ge \left( \frac{n}{2} \right)^2 \lg \left( \frac{n}{2} \right) \ge \left( \frac{n}{2} \right) \frac{n \lg n}{4} = \frac{n^2 \lg n}{8}.$$

It follows that

$$\sum_{i=1}^{n} i \lg i = \Omega(n^2 \lg n).$$

Therefore

$$\sum_{i=1}^{n} i \lg i = \Theta(n^2 \lg n).$$

37. For sufficiently large $n$, $n^k \ge \max\{2, c\}$. Therefore, for sufficiently large $n$,

$$\lg(n^k + c) \le \lg(n^k + n^k) = \lg 2n^k \le \lg n^k n^k = \lg n^{2k} = 2k \lg n.$$

Also, $\lg(n^k + c) \ge \lg n^k = k \lg n$ for all $n$. Therefore, $\lg(n^k + c) = \Theta(\lg n)$.

38.
$$\begin{aligned}
\sum_{i=0}^{k} \lg(n/2^i) &= (k+1)\lg n - (1 + 2 + \cdots + k) \\
&= (k+1)k - \frac{k(k+1)}{2} \\
&= \frac{(k+1)k}{2} \\
&= \frac{(1 + \lg n)\lg n}{2} = \Theta(\lg^2 n)
\end{aligned}$$

40. We show that if $f(n) = \Omega(g(n))$, and $f(n) > 0$ and $g(n) \ge 0$ for all $n \ge 1$, then, for some constant $C$, $f(n) \ge Cg(n)$ for all $n \ge 1$.

*Proof.* Since $f(n) = \Omega(g(n))$, $g(n) = O(f(n))$. By Exercise 39, for some constant $C'$, $g(n) \le C' f(n)$ for all $n \ge 1$. Taking $C = 1/(1 + C')$, we have $Cg(n) \le f(n)$ for all $n \ge 1$.

41. Exercises 39 and 40 show that if $f(n) = \Theta(g(n))$, and $f(n) > 0$ and $g(n) > 0$ for all $n \geq 1$, then, for some constants $C_1$ and $C_2$, $C_1 g(n) \leq f(n) \leq C_2 g(n)$ for all $n \geq 1$.

43. True: $|2 + \sin n| \leq 3 \leq 3|2 + \cos n|$.

45. True

47. False. A counterexample is $f(n) = 1$ for all $n$, and $g(n) = 1 + 1/n$.

48. False. A counterexample is $f(n) = n$, $g(n) = n^2$.

50. True

51. True

54. If $f(n) \neq O(g(n))$, then for every $C > 0$, $|f(n)| > C|g(n)|$ *for infinitely many* $n$. If $g(n) \neq O(f(n))$, then for every $C > 0$, $|g(n)| > C|f(n)|$ *for infinitely many* $n$. However, there is no guarantee that even one $n$ for which $|f(n)| > C|g(n)|$ is true also makes $|g(n)| > C|f(n)|$ true.

55. $f(n) = \begin{cases} 1 & \text{if } n \text{ is even} \\ 0 & \text{if } n \text{ is odd,} \end{cases} \qquad g(n) = 1 - f(n)$

57. We prove the result by using induction on $k$.

    BASIS STEP ($k = 1$). By Exercise 17, Section 2.4, $1 + nx \leq (1 + x)^n$, for $x \geq -1$ and $n \geq 1$. Thus, for $x > 0$ and $n \geq 1$, $nx < (1 + x)^n$ or

$$n < \frac{1}{x}(1 + x)^n.$$

    Taking $C = 1/x$ and $c = 1 + x$ gives the desired result.

    INDUCTIVE STEP. Assume that if $c > 1$, there exists a constant $C$ such that $n^k \leq Cc^n$ for all but finitely many $n$. Let $c > 1$. By the inductive assumption, there exists a constant $C_1$ such that

$$n^k \leq C_1(\sqrt{c})^n$$

    for all but finitely many $n$. By the Basis Step, there exists a constant $C_2$ such that

$$n \leq C_2(\sqrt{c})^n$$

    for all but finitely many $n$. Multiplying these inequalities, we obtain

$$n^{k+1} = n^k n \leq C_1 C_2 (\sqrt{c})^n (\sqrt{c})^n = C_1 C_2 c^n$$

    for all but finitely many n. The Inductive Step is complete. Therefore $n^k = O(c^n)$ for all $k \geq 1$ and $c > 1$.

58. $f(n) = h(n) = t(n) = n$, $g(n) = 2n$

59. The $\Theta$-notation ignores constants that are present in the formula for the *actual* time.

61. Yes

63. By referring to a graph like that of Exercise 62, with $y = 1/x$ replaced by $y = x^m$, we find that

$$1^m + 2^m + \cdots + n^m < \int_1^{n+1} x^m \, dx = \frac{(n+1)^{m+1} - 1}{m+1} < \frac{(n+1)^{m+1}}{m+1}.$$

    The other inequality is proved in a similar manner.

65. We rewrite the inequality of Exercise 64 as

$$b^n[b-(n+1)(b-a)] < a^{n+1}.$$

If we set $a = 1 + 1/(n+1)$ and $b = 1 + 1/n$, the term in brackets reduces to 1 and we have

$$\left(1+\frac{1}{n}\right)^n < \left(1+\frac{1}{n+1}\right)^{n+1}.$$

Therefore, the sequence $\{(1+1/n)^n\}$ is increasing.

66. We rewrite the inequality of Exercise 64 as

$$b^n[b-(n+1)(b-a)] < a^{n+1}.$$

If we set $a = 1$ and $b = 1 + 1/(2n)$, the term in brackets reduces to $\frac{1}{2}$, and we have

$$\left(1+\frac{1}{2n}\right)^n < 2.$$

Squaring both sides gives

$$\left(1+\frac{1}{2n}\right)^{2n} < 4.$$

By Exercise 65, $\{(1+1/n)^n\}$ is increasing; thus,

$$\left(1+\frac{1}{n}\right)^n < \left(1+\frac{1}{2n}\right)^{2n} < 4.$$

68. Using Exercise 67, we have

$$\sum_{i=1}^{n} \frac{1}{i} \le \sum_{i=1}^{n} [\lg(i+1) - \lg i] = \lg(n+1) \le \lg 2n = 1 + \lg n \le 2 \lg n,$$

if $n \ge 2$. Thus,

$$\sum_{i=1}^{n} \frac{1}{i} = O(\lg n).$$

Again, using Exercise 67, we have

$$\sum_{i=1}^{n} \frac{1}{i} > \frac{1}{2} \sum_{i=1}^{n} [\lg(i+1) - \lg i] = \frac{1}{2} \lg(n+1) \ge \frac{1}{2} \lg n.$$

Thus,

$$\sum_{i=1}^{n} \frac{1}{i} = \Omega(\lg n).$$

Therefore,

$$\sum_{i=1}^{n} \frac{1}{i} = \Theta(\lg n).$$

69. We derive the following equivalent inequalities:

$$\begin{aligned}(n+1)^{1/(n+1)} &< n^{1/n}\\ (n+1)^n &< n^{n+1}\\ \frac{(n+1)^n}{n^n} &< n\\ \left(1+\frac{1}{n}\right)^n &< n.\end{aligned}$$

By Exercise 66, $(1+1/n)^n < n$ if $n \geq 4$. It may be directly verified that $(1+1/n)^n < n$ for $n = 3$, and thus the last displayed inequality holds for all $n \geq 3$. Since the displayed inequalities are equivalent, it follows that the first is also true for all $n \geq 3$. Therefore $\{n^{1/n}\}_{n=3}^{\infty}$ is decreasing.

71. Set

$$a = 1 - \frac{1}{n} \quad \text{and} \quad b = 1 - \frac{1}{n+1}$$

to prove that $\{(1-1/n)^n\}_{n=1}^{\infty}$ is increasing. Set

$$a = 1 - \frac{1}{2n} \quad \text{and} \quad b = 1$$

to prove that $\{(1-1/n)^n\}_{n=1}^{\infty}$ is bounded above by 4/9.

72. Notice that

$$\left(1-\frac{1}{n}\right)^n = \left(\frac{n-1}{n}\right)^n.$$

If we replace $n$ by $n+1$, we find using Exercise 71 that the sequence whose general term is

$$\left(\frac{n}{n+1}\right)^{n+1}$$

is increasing. It follows that the sequence whose general term is

$$\frac{1}{\left(\frac{n}{n+1}\right)^{n+1}} = \left(\frac{n+1}{n}\right)^{n+1} = \left(1+\frac{1}{n}\right)^{n+1}$$

is decreasing.

74. In the argument given, the constant is dependent on $n$.

76. False. A counterexample is $f(n) = n$ and $g(n) = n^2$.

77. True. In fact, we can conclude that $f(n) = \Theta(g(n))$ (see the hint to Exercise 78).

79. False. A counterexample is $f(n) = 1$ for all $n$, and

$$g(n) = \begin{cases} 1, & \text{if } n \text{ is even} \\ 2, & \text{if } n \text{ is odd.} \end{cases}$$

80. False. For a counterexample, see the solution to Exercise 79.

81. Inductive Step. Assume that the inequality holds for $n$. Now

$$\begin{aligned}\lg(n+1)! &= \lg(n+1)+\lg n!\\ &\geq \lg(n+1)+\frac{n}{2}\lg\frac{n}{2}.\end{aligned}$$

If we can show that

$$\lg(n+1)+\frac{n}{2}\lg\frac{n}{2}\geq\frac{n+1}{2}\lg\frac{n+1}{2},$$

the inductive step will be complete.

This last inequality is equivalent, in turn, to

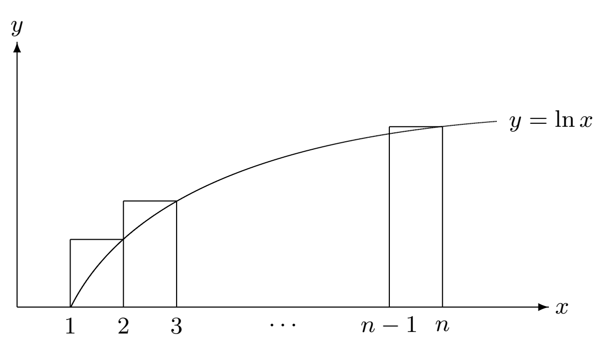
$$\begin{aligned}\lg(n+1)+\frac{n}{2}\lg n-\frac{n}{2} &\geq \frac{n+1}{2}\lg(n+1)-\frac{n+1}{2}\\ \frac{n}{2}\lg n &\geq \frac{n-1}{2}\lg(n+1)-\frac{1}{2}\\ n\lg n &\geq (n-1)\lg(n+1)-1\\ n\lg n &\geq n\lg(n+1)-\lg(n+1)-1\\ 1+\lg(n+1) &\geq n[\lg(n+1)-\lg n]\\ 1+\lg(n+1) &\geq n\left(\lg\frac{n+1}{n}\right)\\ \lg(2(n+1)) &\geq \lg\left(\frac{n+1}{n}\right)^n\\ 2(n+1) &\geq \left(\frac{n+1}{n}\right)^n=\left(1+\frac{1}{n}\right)^n \qquad (4.1)\end{aligned}$$

By Exercise 66,

$$\left(1+\frac{1}{n}\right)^n<4,$$

for all $n$. Thus inequality (4.1) holds for all $n$.

82. In the following figure,




the area of the rectangles exceeds the area under the curve from 1 to $n$, so

$$\int_1^n \ln x\,dx\leq\sum_{k=1}^n \ln k.$$

Now

$$n \ln n - n < n \ln n - n + 1 = \int_1^n \ln x \, dx.$$

84. The inequality may be rewritten

$$\frac{n}{2}[(\lg n) - 1] \le \lg n!.$$

Assuming the result of Exercise 83, it suffices to show that

$$\frac{n}{2}[(\lg n) - 1] \le n \lg n - n \lg e,$$

or, equivalently

$$\frac{1}{2}[(\lg n) - 1] \le \lg n - \lg e,$$

or, equivalently

$$(\lg e) - \frac{1}{2} \quad \le \quad \frac{\lg n}{2}. \tag{4.2}$$

Since

$$(\lg e) - \frac{1}{2} = 1.44\ldots - 0.5 = 0.9\ldots,$$

inequality (4.2) is obviously true for $n \ge 4$. (The given inequality is clearly true for $n = 1, 2, 3$.)

## Problem-Solving Corner: Design and Analysis of an Algorithm

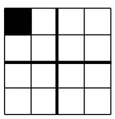
1. Input: $s_1, \ldots, s_n$
 Output: *max*, maximum sum of consecutive values
 *begin_ind*, starting index of values that give the maximum sum, or 0 if every sum is negative
 *end_ind*, ending index of values that give the maximum sum, or 0 if every sum is negative

```
max_sum4(s, n) {
   max = 0
   sum = 0
   begin_ind = 0
   end_ind = 0
   local_begin_ind = 1
   for i = 1 to n {
      if (sum + s_i > 0) {
         local_end_ind = i
         sum = sum + s_i
      }
      else {
         local_begin_ind = i + 1
```

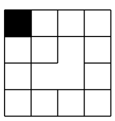
```
            sum = 0
        }
        if (sum > max) {
            begin_ind = local_begin_ind
            end_ind = local_end_ind
            max = sum
        }
    }
}
```

## Section 4.4

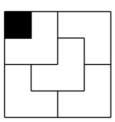
2. Since $n \neq 2$, we proceed to line 6 where we divide the board into four $2 \times 2$ boards:



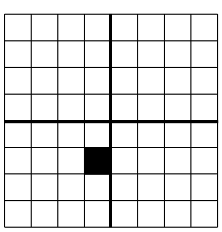
At line 8 we place one right tromino in the center:



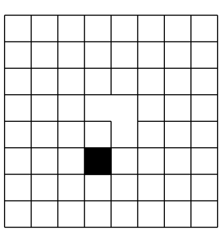
At lines 9 through 12, we recursively tile the $2 \times 2$ boards:



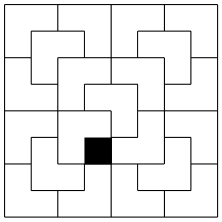
3. Since $n \neq 2$, we proceed to line 6 where we divide the board into four $4 \times 4$ boards:



At line 8 we place one right tromino in the center (we do not show the board rotated):



At lines 9 through 12, we recursively tile the $4 \times 4$ boards:



5. Since $n \neq 1$ and $n \neq 2$, we execute the line

   return $robot\_walk(n-1) + robot\_walk(n-2)$

   with $n = 4$. When we compute $robot\_walk(3)$, since $n \neq 1$ and $n \neq 2$, we execute the line

   return $robot\_walk(n-1) + robot\_walk(n-2)$

   with $n = 3$. The algorithm returns the value 2 for $robot\_walk(2)$ and the value 1 for $robot\_walk(1)$. Therefore for $n = 3$, the algorithm returns the value

   $$robot\_walk(2) + robot\_walk(1) = 2 + 1 = 3.$$

   Therefore for $n = 4$, the algorithm returns the value

   $$robot\_walk(3) + robot\_walk(2) = 3 + 2 = 5.$$

6. Since $n \neq 1$ and $n \neq 2$, we execute the line

   return $robot\_walk(n-1) + robot\_walk(n-2))$

   with $n = 5$. Exercise 5 shows that the algorithm returns the value 5 for $robot\_walk(4)$ and the value 3 for $robot\_walk(3)$. Therefore for $n = 5$, the algorithm returns the value

   $$robot\_walk(4) + robot\_walk(3) = 5 + 3 = 8.$$

8. We use induction on $n$. The Basis Steps, which are readily verified, are $n = 1, 2$. For the Inductive Step, assume that

   $$walk(k) = f_{k+1}$$

   for all $k < n$. From the formula in this section, we have

   $$walk(n) = walk(n-1) + walk(n-2).$$

   By the inductive assumption,

   $$walk(n-1) = f_n \qquad \text{and} \qquad walk(n-2) = f_{n-1}.$$

   Now

   $$walk(n) = walk(n-1) + walk(n-2) = f_n + f_{n-1} = f_{n+1}.$$

9. (a) Input: $n$
   Output: $1 + 2 + \cdots + n$

```
sum(n) {
   if (n == 1)
      return 1
   return n + sum(n − 1)
}
```

(b) Basis Step ($n = 1$). If $n$ is equal to 1, we correctly output 1 and stop.

Inductive Step. Assume that the algorithm correctly computes the sum when the input is $n - 1$. Now suppose that the input to this algorithm is $n > 1$. Since $n \neq 1$, we invoke this procedure with input $n - 1$. By the inductive assumption, the value $v$ returned is equal to

$$1 + \cdots + (n - 1).$$

We then return

$$v + n = 1 + \cdots + (n - 1) + n,$$

which is the correct value.

11. (a) Input: $n$
Output: The number of ways the robot can walk $n$ meters

```
walk3(n) {
   if (n == 1)
      return 1
   if (n == 2)
      return 2
   if (n == 3)
      return 4
   return walk3(n − 1) + walk3(n − 2) + walk3(n − 3)
}
```

(b) Basis Steps ($n = 1, 2, 3$). If $n$ is equal to 1, 2, or 3, we correctly output the number of ways the robot can walk 1, 2, or 3 meters.

Inductive Step. Assume that the algorithm correctly computes the sum when the input is less than $n$. The robot's first step is either 1, 2, or 3 meters. If the first step is 1 meter, the robot must finish its walk by walking $n - 1$ meters. By the inductive assumption the robot can complete the walk in $walk3(n - 1)$ ways. Similarly, if the first step is 2 meters, the robot can complete the walk in $walk3(n - 2)$ ways, and if the first step is 3 meters, the robot can complete the walk in $walk3(n-3)$ ways. Thus the total number of ways the robot can walk $n$ meters is

$$walk3(n - 1) + walk3(n - 2) + walk3(n - 3).$$

Since this is the value computed by the procedure, the algorithm is correct.

12. Input: The sequence $s_1, s_2, \ldots, s_n$ and the length $n$ of the sequence
Output: The minimum value in the sequence

```
find_min(s, n) {
   if (n == 1)
      return s_1
   x = find_min(s, n − 1)
   if (x < s_n)
      return x
   else
      return s_n
}
```

We prove that the algorithm is correct using induction on $n$. The Basis Step is $n = 1$. If $n = 1$, the only item in the sequence is $s_1$ and the algorithm correctly returns it.

Assume that the algorithm computes the minimum for input of size $n$, and suppose that the algorithm receives input of size $n + 1$. By assumption, the recursive call

$x = \mathit{find\_min}(s, n)$

correctly computes $x$ as the minimum value in the sequence $s_1, \ldots, s_n$. If $x$ is less than $s_{n+1}$, the minimum value in the sequence $s_1, \ldots, s_{n+1}$ is $x$—the value returned by the algorithm. If $x$ is not less than $s_{n+1}$, the minimum value in the sequence $s_1, \ldots, s_{n+1}$ is $s_{n+1}$—again, the value returned by the algorithm. In either case, the algorithm correctly computes the minimum value in the sequence. The Inductive Step is complete, and we have proved that the algorithm is correct.

14. Input: The sequence $s_i, \ldots, s_j$, $i$, and $j$
Output: The sequence in reverse order

```
reverse(s, i, j) {
   if (i ≥ j)
      return
   swap(s_i, s_j)
   reverse(s, i + 1, j − 1)
}
```

15. Input: $n$
Output: $n!$

```
factorial(n) {
   fact = 1
   for i = 2 to n
      fact = i * fact
   return fact
}
```

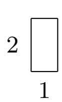
17. To list all of the ways that a robot can walk $n$ meters, set $s$ to the null string and invoke this algorithm.

Input: $n, s$ (a string)
Output: All the ways the robot can walk $n$ meters. Each method of walking $n$ meters includes the extra string $s$ in the list.

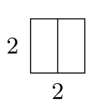
```
list_walk2(n, s) {
   if (n == 1) {
      println(s + " take one step of length 1")
      return
   }
   if (n == 2) {
      println(s + " take two steps of length 1")
      println(s + " take one step of length 2")
      return
   }
   if (n == 3) {
      println(s + " take three steps of length 1")
      println(s + " take one step of length 1 and one step of length 2")
      println(s + " take one step of length 2 and one step of length 1")
      println(s + " take one step of length 3")
      return
   }
   s' = s + " take one step of length 3" // + is concatenation
   list_walk2(n − 3, s')
   s' = s + " take one step of length 2"
   list_walk2(n − 2, s')
   s' = s + " take one step of length 1"
   list_walk2(n − 1, s')
}
```

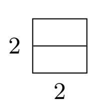
19. 233

20. We use strong induction on $n$. The Basis Steps are $n = 1, 2$. If $n = 1$, there is $1 = f_2$ way to tile a $2 \times 1$ board with $1 \times 2$ rectangular pieces:

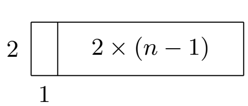



If $n = 2$, there are $2 = f_3$ ways to tile a $2 \times 2$ board with $1 \times 2$ rectangular pieces:

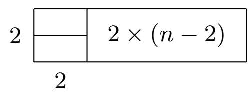



Now suppose that $n > 2$ and if $k < n$, the number of ways to tile a $2 \times k$ board with $1 \times 2$ rectangular pieces is $f_{k+1}$. Now the first two vertical $1 \times 1$ squares can be covered in two ways: using one $1 \times 2$ rectangular piece



or by using two $1 \times 2$ rectangular pieces




By the inductive assumption, in the first case the remaining $2 \times (n-1)$ board can be tiled in $f_n$ ways, and in the second case, the remaining $2 \times (n-2)$ board can be tiled in $f_{n-1}$ ways. Thus the total number of ways to tile a $2 \times n$ board with $1 \times 2$ rectangular pieces is

$$f_n + f_{n-1} = f_{n+1}.$$

The inductive step is complete.

22. $f_{n+2}^2 - f_{n+1}^2 = (f_{n+2} + f_{n+1})(f_{n+2} - f_{n+1}) = f_{n+3} f_n$

23. Using Exercises 21 and 22 and the recurrence relation for the Fibonacci sequence, we have

$$f_{n-2} f_{n+2} = f_{n-2}(f_n + f_{n+1}) = f_n f_{n-2} + f_{n-2} f_{n+1} = f_{n-1}^2 + (-1)^{n+1} + f_n^2 - f_{n-1}^2.$$

The conclusion now follows immediately.

25. Basis Step ($n = 2$). $f_4 = 3 = 4 - 1 = f_3^2 - f_1^2$. $f_5 = 5 = 1 + 4 = f_2^2 + f_3^2$.

Inductive Step. Assume true for $n$. Now

$$\begin{aligned}
f_{2n+2} &= f_{2n+1} + f_{2n} \\
&= f_n^2 + f_{n+1}^2 + f_{n+1}^2 - f_{n-1}^2 \\
&= 2f_{n+1}^2 + f_n^2 - f_{n-1}^2 \\
&= 2f_{n+1}^2 + f_n^2 - (f_{n+1} - f_n)^2 \\
&= 2f_{n+1}^2 + f_n^2 - f_{n+1}^2 + 2f_n f_{n+1} - f_n^2 \\
&= f_{n+1}^2 + 2f_n f_{n+1} \\
&= (f_{n+1} + f_n)^2 - f_n^2 \\
&= f_{n+2}^2 - f_n^2.
\end{aligned}$$

We now use the just proved formula for $f_{2n+2}$ to prove the formula for $f_{2n+3}$:

$$f_{2n+3} = f_{2n+2} + f_{2n+1} = f_{n+2}^2 - f_n^2 + f_n^2 + f_{n+1}^2 = f_{n+1}^2 + f_{n+2}^2.$$

The inductive step is complete.

26. Basis Steps ($n = 1, 2$). $f_1$ and $f_2$ are both odd ($f_1 = f_2 = 1$) and neither 1 nor 2 is divisible by 3. Therefore, the statement is true for $n = 1, 2$.

Inductive Step. Assume that the statement is true for all $k < n$. We must prove that the statement is true for $n$. We can assume that $n > 2$. We consider two cases: $n$ is divisible by 3 and $n$ is not divisible by 3.

If $n$ is divisible by 3, then neither $n-1$ nor $n-2$ is divisible by 3. By the inductive assumption, both $f_{n-1}$ and $f_{n-2}$ are odd. Since $f_n = f_{n-1} + f_{n-2}$, $f_n$ is even. Therefore, if $n$ is divisible by 3, $f_n$ is even.

If $n$ is not divisible by 3, then exactly one of $n-1$ or $n-2$ is divisible by 3. By the inductive assumption, one of $f_{n-1}$ and $f_{n-2}$ is odd and the other is even. Since $f_n = f_{n-1} + f_{n-2}$, $f_n$ is odd. Therefore, if $n$ is not divisible by 3, $f_n$ is odd.

We have shown that $f_n$ is even if and only if $n$ is divisible by 3, so the Inductive Step is complete.

28. Basis Steps ($n = 1, 2$). $f_1 = 1 \leq 1 = 2^0$; $f_2 = 1 \leq 2 = 2^1$

Inductive Step. Assume that the equation is true for $n-2$ and $n-1$. Now

$$f_n = f_{n-1} + f_{n-2} \leq 2^{n-2} + 2^{n-3} < 2^{n-2} + 2^{n-2} = 2^{n-1}.$$

29. Basis Steps ($n = 1, 2$). For $n = 1$, we have $f_2 = 1 = 2 - 1 = f_3 - 1$, and $f_1 = 1 = f_2$.

For $n = 2$, we have $f_2 + f_4 = 1 + 3 = 5 - 1 = f_5 - 1$, and $f_1 + f_3 = 1 + 2 = 3 = f_4$.

Inductive Step.

$$\begin{aligned} \sum_{k=1}^{n+1} f_{2k} &= \sum_{k=1}^{n} f_{2k} + f_{2n+2} \\ &= f_{2n+1} - 1 + f_{2n+2} = f_{2n+3} - 1 \\ \sum_{k=1}^{n+1} f_{2k-1} &= \sum_{k=1}^{n} f_{2k-1} + f_{2n+1} \\ &= f_{2n} + f_{2n+1} = f_{2n+2} \end{aligned}$$

31. We show that the representation

$$n = \sum_{i=1}^{m} f_{k_i} \tag{4.3}$$

given in the hint for Exercise 30 is unique.

By Exercise 29, the partial sum of Fibonacci numbers with even indexes is

$$\sum_{k=1}^{n} f_{2k} = f_{2n+1} - 1.$$

If we do not allow $f_1$ as a summand, the partial sum of Fibonacci numbers with odd indexes becomes

$$\sum_{k=2}^{n} f_{2k-1} = \sum_{k=1}^{n} f_{2k-1} - f_1 = f_{2n} - 1,$$

where we have again used Exercise 29.

Suppose by way of contradiction that some integer has a representation as the sum of distinct Fibonacci numbers no two of which are consecutive different from (4.3). Let $n$ denote the smallest such integer. Let

$$n = \sum_{i=1}^{j} f_{t_i},$$

where $t_1 > t_2 > \ldots$, be another representation of $n$ as the sum of distinct Fibonacci numbers no two of which are consecutive.

Since $f_{t_1} \le n$ and $f_{k_1}$ is the largest Fibonacci number less than or equal to $n$, $f_{t_1} \le f_{k_1}$. If $f_{t_1} = f_{k_1}$, $n - f_{k_1}$ has at least two representations as the sum of distinct Fibonacci numbers no two of which are consecutive, which contradicts the minimality of $n$. Therefore $f_{t_1} < f_{k_1}$. Thus

$$f_{t_1} \le f_{k_1-1}.$$

In the representation, no two Fibonacci numbers are consecutive, thus

$$f_{t_2} \le f_{k_1-3}, \; f_{t_3} \le f_{k_1-5}, \; \ldots$$

Therefore

$$\begin{aligned} n &= f_{t_1} + f_{t_2} + \cdots + f_{t_j} \\ &\le f_{k_1-1} + f_{k_1-3} + \cdots + f_{k_1-(2j-1)} \\ &\le f_{k_1-1} + f_{k_1-3} + \cdots + f_p && \text{where } p = 2 \text{ or } 3 \\ &= f_{k_1} - 1 && \text{by the preceding comments} \\ &< n, \end{aligned}$$

which is a contradiction. Therefore the representation of an integer as the sum of distinct Fibonacci numbers no two of which are consecutive is unique if we do not allow $f_1$ as a summand.

32. Exercise 21 shows that

$$f_n^2 = f_{n-1}f_{n+1} + (-1)^{n+1}, \quad n \ge 2,$$

so

$$f_n^2 = f_{n-1}(f_n + f_{n-1}) + (-1)^{n+1}, \quad n \ge 2$$

or

$$f_n^2 - f_{n-1}f_n - f_{n-1}^2 - (-1)^{n+1} = 0, \quad n \ge 2.$$

The quadratic formula gives

$$\begin{aligned} f_n &= \frac{f_{n-1} \pm \sqrt{f_{n-1}^2 - 4[-f_{n-1}^2 - (-1)^{n+1}]}}{2} \\ &= \frac{f_{n-1} \pm \sqrt{5f_{n-1}^2 + 4(-1)^{n+1}}}{2}, \quad n \ge 2. \end{aligned}$$

The negative sign gives an extraneous root since if chosen, we would have, for $n \ge 3$,

$$f_n < \frac{f_{n-1}}{2}$$

or

$$2(f_{n-1} + f_{n-2}) < f_{n-1}$$

or

$$f_{n-1} + 2f_{n-2} < 0$$

which is impossible. The formula for $n = 2$ is correct by inspection.

34. Basis Steps ($n = 3, 4$). For $n = 3$, we have

$$g_1 f_1 + g_2 f_2 = g_1 + g_2 = g_3.$$

For $n = 4$, we have

$$g_1 f_2 + g_2 f_3 = g_1 + 2g_2 = g_2 + (g_1 + g_2) = g_2 + g_3 = g_4.$$

Inductive Step. Assume that $n > 4$. Now

$$\begin{aligned} g_n = g_{n-1} + g_{n-2} &= (g_1 f_{n-3} + g_2 f_{n-2}) + (g_1 f_{n-4} + g_2 f_{n-3}) \\ &= g_1(f_{n-3} + f_{n-4}) + g_2(f_{n-2} + f_{n-3}) \\ &= g_1 f_{n-2} + g_2 f_{n-1}. \end{aligned}$$

36. The formula reduces the problem of integrating $\log^n |x|$ to the problem of integrating $\log^{n-1} |x|$, a simpler instance of the original problem. Eventually the problem is reduced to integrating $\log |x|$, which is straightforward.

Another example of a recursive integration formula is

$$\int \sin^{2n} x\, dx = -\frac{\sin^{2n-1} x \cos x}{2n} + \frac{2n-1}{2n} \int \sin^{2n-2} x\, dx.$$

# Chapter 5

# Solutions to Selected Exercises

## Section 5.1

2. Since $\lfloor\sqrt{47}\rfloor = 6$, the for loop in Algorithm 5.1.8 runs from $d = 2$ to 6. For each of these values, $n \bmod d \neq 0$. Therefore the algorithm returns 0 to signal that 47 is prime.

3. Since $\lfloor\sqrt{209}\rfloor = 14$, the for loop in Algorithm 5.1.8 runs from $d = 2$ to 14. For $d = 2$ to 10, $n \bmod d \neq 0$. When $d = 11$, $n \bmod d = 0$. Therefore the algorithm returns 11 to signal that 209 is composite and 11 is a divisor of 209.

5. Since $\lfloor\sqrt{1007}\rfloor = 31$, the for loop in Algorithm 5.1.8 runs from $d = 2$ to 31. For $d = 2$ to 18, $n \bmod d \neq 0$. When $d = 19$, $n \bmod d = 0$. Therefore the algorithm returns 19 to signal that 1007 is composite and 19 is a divisor of 1007.

6. Since $\lfloor\sqrt{4141}\rfloor = 64$, the for loop in Algorithm 5.1.8 runs from $d = 2$ to 64. For $d = 2$ to 40, $n \bmod d \neq 0$. When $d = 41$, $n \bmod d = 0$. Therefore the algorithm returns 41 to signal that 4141 is composite and 41 is a divisor of 4141.

8. Since $\lfloor\sqrt{1050703}\rfloor = 1025$, the for loop in Algorithm 5.1.8 runs from $d = 2$ to 1025. For $d = 2$ to 100, $n \bmod d \neq 0$. When $d = 101$, $n \bmod d = 0$. Therefore the algorithm returns 101 to signal that 1050703 is composite and 101 is a divisor of 1050703.

10. (For Exercise 1) $9 = 3 \cdot 3$

11. $$\begin{aligned} 11! &= 11 \cdot 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \\ &= 11 \cdot (5 \cdot 2) \cdot (3 \cdot 3) \cdot (2 \cdot 2 \cdot 2) \cdot 7 \cdot (3 \cdot 2) \cdot 5 \cdot (2 \cdot 2) \cdot 3 \cdot 2 \\ &= 2^8 3^4 5^2 7^1 11^1 \end{aligned}$$

13. 5

14. 30

16. 20

17. 15

19. 331

20. 1

22. 15

23. 7

26. (For Exercise 13) We have $\gcd(5, 25) = 5$ and $\operatorname{lcm}(5, 25) = 25$. Thus

$$\gcd(5, 25) \cdot \operatorname{lcm}(5, 25) = 5 \cdot 25.$$

27. Since $d \mid m$, $m = dq_1$ for some integer $q_1$. Since $d \mid n$, $n = dq_2$ for some integer $q_2$. Now

$$m - n = dq_1 - dq_2 = d(q_1 - q_2).$$

Therefore, $d \mid (m - n)$.

29. Since $d_1 \mid m$, $m = d_1q_1$ for some integer $q_1$. Since $d_2 \mid n$, $n = d_2q_2$ for some integer $q_2$. Now

$$mn = (d_1q_1)(d_2q_2) = (d_1d_2)(q_1q_2).$$

Therefore, $d_1d_2 \mid mn$.

30. Since $dc \mid nc$, $nc = dcq$ for some integer $q$. Since $dc$ is a divisor of $nc$, $c \neq 0$. Therefore, we may cancel $c$ in $nc = dcq$ to obtain $n = dq$. Therefore, $d \mid n$.

32. After checking whether 2 is a divisor of $n$, we need not check whether $4, 6, 8, \ldots$ divide $n$. Implementing this change cuts the time by about one-half:

```
is_prime(n) {
   if (n mod 2 == 0)
      return 2
   d = 3
   while (d ≤ ⌊√n⌋) {
      if (n mod d == 0)
         return d
      d = d + 2
   }
   return 0
}
```

Continuing this idea, if we store the primes less than or equal to $\lfloor\sqrt{n}\rfloor$, we need only check whether any of these primes divides $n$.

33. $2 \cdot 3 \cdot 5 \cdot 7 \cdot 11 \cdot 13 + 1 = 30031 = 509 \cdot 59$

34. Inductive Step. Assume that $\{a_1, \ldots, a_n\}$ is a *-set. Let $b_0 = \prod_{k=1}^{n} a_k$ and $b_i = b_0 + a_i$ for $1 \leq i \leq n$. We prove that $\{b_0, b_1, \ldots, b_n\}$ is a *-set.

We first show that $(b_0 - b_j) \mid b_0$ for all $j \geq 1$. Since $b_0 = \prod_{k=1}^{n} a_k$, $b_0 - b_j = -a_j$, and $j \geq 1$, it is clear that $(b_0 - b_j) \mid b_0$.

We next show that $(b_j - b_0) \mid b_j$ for all $j \geq 1$. Since $b_j - b_0 = a_j$ and $b_j = b_0 + a_j$, we must show that $a_j \mid (b_0 + a_j)$. Since $b_0 = \prod_{k=1}^{n} a_k$ and $j \geq 1$, $a_j \mid b_0$. Trivially, $a_j \mid a_j$; therefore, $a_j \mid (b_0 + a_j)$.

Finally, we show that for all $i, j > 0$, $(b_i - b_j) \mid b_i$. Since

$$b_i - b_j = (b_0 + a_i) - (b_0 + a_j) = a_i - a_j$$

and

$$b_i = b_0 + a_i,$$

it suffices to show that $(a_i - a_j) \mid (b_0 + a_i)$. By the inductive assumption, $(a_i - a_j) \mid a_i$. Since $b_0 = \prod_{k=1}^{n} a_k$ and $i \geq 1$, $a_i \mid b_0$. Thus $(a_i - a_j) \mid b_0$. Therefore $(a_i - a_j) \mid (b_0 + a_i)$ and the Inductive Step is complete.

35. $\{2, 3, 4\}$, $\{24, 26, 27, 28\}$

## Section 5.2

2. 6  3. 7  5. 8  6. 1001  9. 27  10. 219

12. 255  13. 3547  15. 111101  16. 11011111  18. 10000000000

19. 11000000110100  21. 101000  22. 1100011  24. 1000100010

25. 100010100  27. 489  28. 15996  30. 8349  31. 307322

33. (For Exercise 14) 22  34. (For Exercise 26) 111010

36. 903  37. 565D  39. 130FF7

41. 1101010 represents a number in binary, decimal, and hexadecimal.

43. 4003  44. 4041  46. 519  47. 179889

49. (For Exercise 14) 42  50. (For Exercise 26) 72  52. Yes

53. 30470 does not represent a number in binary, but it does represent a number in octal, decimal, and hexadecimal.

55. Suppose that the base $b$ representation of $m$ is

$$m = \sum_{i=0}^{k} c_i b^i,$$

$c_k \neq 0$. Then

$$b^k \leq c_k b^k \leq m,$$

and

$$m \leq \sum_{i=0}^{k} (b-1)b^i = (b-1)\sum_{i=0}^{k} b^i = (b-1)\frac{b^{k+1}-1}{b-1} = b^{k+1} - 1 < b^{k+1}.$$

Since $b^k \leq m$, taking logs to the base $b$, we obtain

$$k \leq \log_b m.$$

Since $m < b^{k+1}$, again taking logs, we obtain

$$\log_b m < k + 1.$$

Combining these inequalities, we have

$$k + 1 \leq 1 + \log_b m < k + 2.$$

Therefore, the number of digits required to represent $m$ is

$$k + 1 = \lfloor 1 + \log_b m \rfloor.$$

57. The algorithm begins by setting *result* to 1 and $x$ to $a$. Since $n = 15 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = 1 * a = a$. $x$ becomes $a^2$, and $n$ becomes 7.

    Since $n = 7 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = a * a^2 = a^3$. $x$ becomes $a^4$, and $n$ becomes 3.

    Since $n = 3 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = a^3 * a^4 = a^7$. $x$ becomes $a^8$, and $n$ becomes 1.

    Since $n = 1 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = a^7 * a^8 = a^{15}$. $x$ becomes $a^{16}$, and $n$ becomes 0.

    Since $n = 0$ is not greater than 0, the while loop terminates. The algorithm returns *result*, which is equal to $a^{15}$.

58. The algorithm begins by setting *result* to 1 and $x$ to $a$. Since $n = 80 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ becomes $a^2$, and $n$ becomes 40.

    Since $n = 40 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ becomes $a^4$, and $n$ becomes 20.

    Since $n = 20 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ becomes $a^8$, and $n$ becomes 10.

    Since $n = 10 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ becomes $a^{16}$, and $n$ becomes 5.

    Since $n = 5 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = 1 * a^{16} = a^{16}$. $x$ becomes $a^{32}$, and $n$ becomes 2.

    Since $n = 2 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ becomes $a^{64}$, and $n$ becomes 1.

    Since $n = 1 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* becomes $result * x = a^{16} * a^{64} = a^{80}$. $x$ becomes $a^{128}$, and $n$ becomes 0.

    Since $n = 0$ is not greater than 0, the while loop terminates. The algorithm returns *result*, which is equal to $a^{80}$.

60. The algorithm begins by setting *result* to 1 and $x$ to $a \bmod z = 143 \bmod 230 = 143$. Since $n = 10 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (143 * 143) \bmod 230 = 20449 \bmod 230 = 209$, and $n$ is set to 5.

    Since $n = 5 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* is set to $(result * x) \bmod z = 209 \bmod 230 = 209$. $x$ is set to $(x * x) \bmod z = (209 * 209) \bmod 230 = 43681 \bmod 230 = 211$, and $n$ is set to 2.

Since $n = 2 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (211 * 211) \bmod 230 = 44521 \bmod 230 = 131$, and $n$ is set to 1.

Since $n = 1 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* is set to $(\mathit{result} * x) \bmod z = (209 * 131) \bmod 230 = 27379 \bmod 230 = 9$. $x$ is set to $(x * x) \bmod z = (131 * 131) \bmod 230 = 17161 \bmod 230 = 141$, and $n$ is set to 0.

Since $n = 0$ is not greater than 0, the while loop terminates. The algorithm returns *result*, which is equal to $a^n \bmod z = 143^{10} \bmod 230 = 9$.

61. The algorithm begins by setting *result* to 1 and $x$ to $a \bmod z = 143 \bmod 230 = 143$. Since $n = 100 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (143 * 143) \bmod 230 = 20449 \bmod 230 = 209$, and $n$ is set to 50.

    Since $n = 50 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (209 * 209) \bmod 230 = 43681 \bmod 230 = 211$, and $n$ is set to 25.

    Since $n = 25 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* is set to $(\mathit{result} * x) \bmod z = 211 \bmod 230 = 211$. $x$ is set to $(x * x) \bmod z = (211 * 211) \bmod 230 = 44521 \bmod 230 = 131$, and $n$ is set to 12.

    Since $n = 12 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (131 * 131) \bmod 230 = 17161 \bmod 230 = 141$, and $n$ is set to 6.

    Since $n = 6 > 0$, the body of the while loop executes. Since $n \bmod 2$ is not equal to 1, *result* is not modified. $x$ is set to $(x * x) \bmod z = (141 * 141) \bmod 230 = 19881 \bmod 230 = 101$, and $n$ is set to 3.

    Since $n = 3 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* is set to $(\mathit{result} * x) \bmod z = (211 * 101) \bmod 230 = 21311 \bmod 230 = 151$. $x$ is set to $(x * x) \bmod z = (101 * 101) \bmod 230 = 10201 \bmod 230 = 81$, and $n$ is set to 1.

    Since $n = 1 > 0$, the body of the while loop executes. Since $n \bmod 2$ is equal to 1, *result* is set to $(\mathit{result} * x) \bmod z = (151 * 81) \bmod 230 = 12231 \bmod 230 = 41$. $x$ is set to $(x * x) \bmod z = (81 * 81) \bmod 230 = 6561 \bmod 230 = 121$, and $n$ is set to 0.

    Since $n = 0$ is not greater than 0, the while loop terminates. The algorithm returns *result*, which is equal to $a^n \bmod z = 143^{100} \bmod 230 = 41$.

63. Basis Step ($n = 1$). Since $T_{n!} = 0$ and $S_n = 1$, the result holds for $n = 1$.

    Inductive Step. Assume true for $n$. Now

$$T_{(n+1)!} = T_{(n+1)n!} = T_{n+1} + T_{n!} = T_{n+1} + n - S_n,$$

where we have used the result of Exercise 62 and the inductive assumption. Thus the inductive step will be complete if we can show that $T_{n+1} = 1 + S_n - S_{n+1}$.

Suppose that $n$ is even. Then $n+1$ is odd, so $T_{n+1} = 0$. Now $n = \ldots 0$ in binary, so $n+1 = \ldots 1$. Therefore $S_{n+1} = 1 + S_n$. Thus $T_{n+1} = 1 + S_n - S_{n+1}$, if $n$ is even.

Suppose that $n$ is odd, say,

$$n = \ldots 0\underbrace{11\ldots 11}_{k\ 1\text{'s}}.$$

Then

$$n+1 = \ldots 1\underbrace{00\ldots 00}_{k\ 0\text{'s}}.$$

Thus $S_n - S_{n+1} = k-1$ and $T_{n+1} = k$. Therefore $T_{n+1} = 1 + S_n - S_{n+1}$, if $n$ is odd.

64. Input: $b, m, b', n$
Output: *prod*, the binary product of $b$ and $b'$

$add(p+q, i, r)$ computes the sum of the binary numbers $p$ and $q$, except that $q$ is shifted $i$ places left (effectively appending $i$ zeros on the right), with the result in $r$.

```
binary_product(b, m, b', n) {
   prod = 0
   for i = 1 to n
      if (b'_i == 1)
         add(prod + b, i, prod)
   return prod
}
```

65. The for loop runs in time $O(n)$. Adding *prod* and $b$ takes time $m$ (the appended zeros are *not* added). Thus the time is $O(mn)$. Since $m$ and $n$ are the number of bits to represent the numbers that are multiplied, the result follows.

## Section 5.3

2. 1 3. 20 5. 20 6. 331 8. 495 9. 23 12. $89, 55$

13. Suppose that lines 3 and 4 in Algorithm 5.3.3 are deleted. If $a \geq b$, clearly the result is the same as in the original form. If $a < b$, then $b \neq 0$ and the first iteration of the while loop swaps $a$ and $b$. Thereafter, the algorithm proceeds as in the original form, and again the result is the same.

15. Let $m$ be a common divisor of $a$ and $b$. By Theorem 5.1.3(a), $m$ divides $a+b$. Thus $m$ is a common divisor of $a$ and $a+b$.

Let $m$ be a common divisor of $a$ and $a+b$. By Theorem 5.1.3(b), $m$ divides $(a+b)-a = b$. Thus $m$ is a common divisor of $a$ and $b$.

Since the set of common divisors of $a$ and $a+b$ is equal to the set of common divisors of $a$ and $b$, $\gcd(a, b) = \gcd(a, a+b)$.

16. Let $m$ be a common divisor of $a$ and $b$. By Theorem 5.1.3(b), $m$ divides $a-b$. Thus $m$ is a common divisor of $b$ and $a-b$.

Let $m$ be a common divisor of $b$ and $a-b$. By Theorem 5.1.3(a), $m$ divides $(a-b)+b = a$. Thus $m$ is a common divisor of $a$ and $b$.

Since the set of common divisors of $b$ and $a-b$ is equal to the set of common divisors of $a$ and $b$, $\gcd(a,b) = \gcd(a-b,b)$.

18. The algorithm given in the hint in the book to Exercise 17 does $m$ subtractions in the worst case, which occurs when $a = m$ and $b = 1$.

19.

| $a$ \ $b$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| 3 | 0 | 1 | 2 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 |
| 4 | 0 | 1 | 1 | 2 | 1 | 2 | 2 | 3 | 1 | 2 | 2 | 3 | 1 | 2 | 2 | 3 | 1 | 2 | 2 | 3 | 1 | 2 |
| 5 | 0 | 1 | 2 | 3 | 2 | 1 | 2 | 3 | 4 | 3 | 1 | 2 | 3 | 4 | 3 | 1 | 2 | 3 | 4 | 3 | 1 | 2 |
| 6 | 0 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 2 | 3 | 3 | 1 | 2 | 2 | 2 | 3 | 3 | 1 | 2 | 2 | 2 |
| 7 | 0 | 1 | 2 | 2 | 3 | 3 | 2 | 1 | 2 | 3 | 3 | 4 | 4 | 3 | 1 | 2 | 3 | 3 | 4 | 4 | 3 | 1 |
| 8 | 0 | 1 | 1 | 3 | 1 | 4 | 2 | 2 | 1 | 2 | 2 | 4 | 2 | 5 | 3 | 3 | 1 | 2 | 2 | 4 | 2 | 5 |
| 9 | 0 | 1 | 2 | 1 | 2 | 3 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 3 | 4 | 3 | 4 | 3 | 1 | 2 | 3 | 2 |
| 10 | 0 | 1 | 1 | 2 | 2 | 1 | 3 | 3 | 2 | 2 | 1 | 2 | 2 | 3 | 3 | 2 | 4 | 4 | 3 | 3 | 1 | 2 |
| 11 | 0 | 1 | 2 | 3 | 3 | 2 | 3 | 4 | 4 | 3 | 2 | 1 | 2 | 3 | 4 | 4 | 3 | 4 | 5 | 5 | 4 | 3 |
| 12 | 0 | 1 | 1 | 1 | 1 | 3 | 1 | 4 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 4 | 2 | 5 | 3 | 3 |
| 13 | 0 | 1 | 2 | 2 | 2 | 4 | 2 | 3 | 5 | 3 | 3 | 3 | 2 | 1 | 2 | 3 | 3 | 3 | 5 | 3 | 4 | 6 |
| 14 | 0 | 1 | 1 | 3 | 2 | 3 | 2 | 1 | 3 | 4 | 3 | 4 | 2 | 2 | 1 | 2 | 2 | 4 | 3 | 4 | 3 | 2 |
| 15 | 0 | 1 | 2 | 1 | 3 | 1 | 2 | 2 | 3 | 3 | 2 | 4 | 2 | 3 | 2 | 1 | 2 | 3 | 2 | 4 | 2 | 3 |
| 16 | 0 | 1 | 1 | 2 | 1 | 2 | 3 | 3 | 1 | 4 | 4 | 3 | 2 | 3 | 2 | 2 | 1 | 2 | 2 | 3 | 2 | 3 |
| 17 | 0 | 1 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 4 | 4 | 4 | 3 | 4 | 3 | 2 | 1 | 2 | 3 | 4 | 3 |
| 18 | 0 | 1 | 1 | 1 | 2 | 4 | 1 | 4 | 2 | 1 | 3 | 5 | 2 | 5 | 3 | 2 | 2 | 2 | 1 | 2 | 2 | 2 |
| 19 | 0 | 1 | 2 | 2 | 3 | 3 | 2 | 4 | 4 | 2 | 3 | 5 | 5 | 3 | 4 | 4 | 3 | 3 | 2 | 1 | 2 | 3 |
| 20 | 0 | 1 | 1 | 3 | 1 | 1 | 2 | 3 | 2 | 3 | 1 | 4 | 3 | 4 | 3 | 2 | 2 | 4 | 2 | 2 | 1 | 2 |
| 21 | 0 | 1 | 2 | 1 | 2 | 2 | 2 | 1 | 5 | 2 | 2 | 3 | 3 | 6 | 2 | 3 | 3 | 3 | 2 | 3 | 2 | 1 |

| $a$ | $b$ | $n$ (= number of modulus operations) |
|---|---|---|
| 1 | 0 | 0 |
| 2 | 1 | 1 |
| 3 | 2 | 2 |
| 5 | 3 | 3 |
| 8 | 5 | 4 |
| 13 | 8 | 5 |
| 21 | 13 | 6 |

21. Using induction on $n$, we prove that when the pair $f_{n+2}$, $f_{n+1}$ is input to the Euclidean algorithm, exactly $n$ modulus operations are required.

BASIS STEP ($n = 1$). Table 5.3.2 shows that when the pair $f_3, f_2$ is input to the Euclidean algorithm, one modulus operation is required.

INDUCTIVE STEP. Assume that when the pair $f_{n+2}$, $f_{n+1}$ is input to the Euclidean algorithm, $n$ modulus operations are required. We must show that when the pair $f_{n+3}, f_{n+2}$ is input to the Euclidean algorithm, $n+1$ modulus operations are required.

At line 6, since

$$f_{n+3} = f_{n+2} + f_{n+1},$$

$r = f_{n+1}$. The algorithm then repeats using the values of $f_{n+2}$ and $f_{n+1}$. By the inductive assumption, exactly $n$ additional modulus operations are required. Thus a total of $n+1$ modulus operations are required.

22. We prove this result by induction on $\max(a, b)$. We omit the Basis Step.

    Without loss of generality, we assume that $a \geq b$. If $b = 0$, zero modulus operations are required to compute $\gcd(a, b)$ and $\gcd(ka, kb)$. Suppose that $b > 0$. Suppose that when we divide $a$ by $b$, we obtain
    $$a = bq + r, \quad 0 \leq r < b.$$
    Multiplying these last inequalities by $k$, we obtain
    $$ka = (kb)q + kr, \quad 0 \leq kr < kb.$$
    Thus when we divide $ka$ by $kb$, we obtain the remainder $kr$. In computing $\gcd(a, b)$, we continue by computing $\gcd(b, r)$. In computing $\gcd(ka, kb)$, we continue by computing $\gcd(kb, kr)$. By the inductive assumption, these computations require the same number of modulus operations. Thus the number of modulus operations required to compute $\gcd(a, b)$ is the same as the number of modulus operations required to compute $\gcd(ka, kb)$. The Inductive Step is complete.

24. By Theorem 5.3.7, there exist integers $s$ and $t$ such that $\gcd(a, b) = sa + tb$. Since $d \mid a$ and $d \mid b$, $d \mid (sa + tb)$. Therefore $d \mid \gcd(a, b)$.

25. If $p$ divides $a$, we are done; so suppose that $p$ does not divide $a$. We must show that $p$ divides $b$. Since $p$ is prime, $\gcd(p, a) = 1$. By Theorem 5.3.7, there are integers $s$ and $t$ such that
    $$1 = sp + ta.$$
    Multiplying both sides of this equation by $b$, we obtain
    $$b = spb + tab.$$
    By Theorem 5.1.3(c), $p$ divides $spb$ and $p$ divides $tab$. By Theorem 5.1.3(a), $p$ divides $spb + tab = b$.

26. $p = 4, \; a = 6, \; b = 10$

29. $X$ has a least element by the Well-Ordering Property.

30. Let $g = sa + tb$. If $c \mid a$ and $c \mid b$, then $c \mid sa + tb = g$.

32. Exercise 31 shows that $g$ is a common divisor. Exercise 30 shows that $g$ is the greatest common divisor.

34. $s = 1$ 35. $s = 3$ 37. $s = 134$ 38. $s = 67$

41. The subsection *Computing an Inverse Modulo an Integer* shows that if $\gcd(n, \phi) = 1$, then $n$ has an inverse modulo $\phi$.

    Now suppose that $n$ has an inverse $s$ modulo $\phi$. Then
    $$ns \bmod \phi = 1.$$
    Since 1 is the remainder, there exists $q$ such that
    $$ns = \phi q + 1.$$
    Now suppose that $c$ is a positive common divisor of $n$ and $\phi$. Then $c$ divides $ns$ and $\phi q$ and also
    $$ns - \phi q = 1.$$
    Therefore $c = 1$ and $\gcd(n, \phi) = 1$.

## Problem-Solving Corner: Making Postage

1. Let $k = \gcd(p, q)$. Suppose that we can make $m$ cents postage using $a$ $p$-cent and $b$ $q$-cent stamps. Then $ap + bq = m$. Then $k \mid m$. Since $k > 1$, $k \nmid m + 1$. Therefore we cannot make $m + 1$ cents postage. The conclusion follows.

## Section 5.4

2. `DRINK YOUR OVALTINE`

3. `WEISERKYEIEFTKK␣`

5. $c = a^n \bmod z = 333^{29} \bmod 713 = 306$

6. $a = c^s \bmod z = 411^{569} \bmod 713 = 500$

8. $\phi = (p-1)(q-1) = 16 \cdot 22 = 352$

9. $s = 159$

11. $a = c^s \bmod z = 250^{159} \bmod 391 = 10$

13. $\phi = (p-1)(q-1) = 58 \cdot 100 = 5800$

14. $s = 3961$

16. $a = c^s \bmod z = 250^{3961} \bmod 5959 = 5648$

# Chapter 6

# Solutions to Selected Exercises

## Section 6.1

2. $2 \cdot 3 \cdot 5$  3. $3 \cdot 3 \cdot 5$  5. $5 \cdot 6 \cdot 2 \cdot 3 \cdot 3$  6. $2^6 - 1$

8. $26^3 10^2,\ 26 \cdot 25 \cdot 24 \cdot 10 \cdot 9$

9. Yes, the ad is correct; there are $5 \cdot 14 \cdot 3 = 210$ dinners.

11. $2 + 3$  12. $2 + 4$  14. $\underbrace{n + n + \cdots + n}_{m\ n\text{'s}} = mn$  15. $32 + 16$

17. $3 + 6$  18. $3 + 3$  21. $6$  22. $6 + 2$  24. $10$  25. $11$

27. $2 \cdot 3^2$  29. $50^2$  30. $50 \cdot 49$  32. $2^6$  33. $3 \cdot 2^6$  35. $(8 \cdot 7)/2$

36. $2^8 - 1$  39. $4 \cdot 3 \cdot 2$  40. $3 \cdot 2 \cdot 4$  42. $5 \cdot 4 + 5 \cdot 4 \cdot 3$  43. $2 \cdot 5 \cdot 4$

45. $5 \cdot 4 \cdot 3$  46. $5^2$  48. $4^3$  49. $4 \cdot 3 \cdot 2$  51. $3 \cdot 4 \cdot 3$

53. $(200 - 4)/2$  54. $(200 - 4)/2$  56. $200 - 72$  57. $5 + 9 \cdot 9 + 9 \cdot 8$

59. $196 - (9 + 2 + 18)$  60. $1 + 1 \cdot 9 \cdot 9 - 2$  62. $2 + 3 + \cdots + 9 = 44$

63. (a) $12 \cdot 11 \cdot 10 \cdot 9 \cdot 8$  (b) $12^5$  (c) $12^5 - 12 \cdot 11 \cdot 10 \cdot 9 \cdot 8$

65. $(5!)(2!)(3!)$  66. $(5!)(5!)$  68. $8! \cdot 9 \cdot 8$

69. $26 + 26 \cdot 36 + 26 \cdot 36^2 + 26 \cdot 36^3 + 26 \cdot 36^4 + 26 \cdot 36^5$  70. $m^n$  72. $2 \cdot 3 \cdot 3 \cdot 2$

73. A subset $X$ has $n$ elements or less if and only if $\overline{X}$ has more than $n$ elements. Thus exactly half of the subsets have $n$ elements or less. Therefore, the number of subsets is $\frac{1}{2} 2^{2n+1} = 2^{2n}$.

75. $2^{21}(2^8 - 2)$  76. $(2^7 - 1)(2^{24} - 2) + 2^{14}(2^{16} - 2) + 2^{21}(2^8 - 2)$

78. We count the number of antisymmetric relations on $\{1, 2, \ldots, n\}$ by computing the number of ways to construct the $n \times n$ matrix of an antisymmetric relation.

    Each element of the diagonal can be either 0 or 1. Thus there are $2^n$ ways to assign values to the diagonal.

For $i$ and $j$ satisfying $1 \leq i < j \leq n$, we can assign the entries in row $i$, column $j$ and row $j$, column $i$ in three ways:

| *Row* $i$, *Column* $j$ | *Row* $j$, *Column* $i$ |
|---|---|
| 0 | 0 |
| 1 | 0 |
| 0 | 1 |

Since there are $(n^2 - n)/2$ values of $i$ and $j$ satisfying $1 \leq i < j \leq n$, we can assign the off-diagonal values in $3^{(n^2-n)/2}$ ways. Therefore, there are

$$2^n \cdot 3^{(n^2-n)/2}$$

antisymmetric relations on an $n$-element set.

79. We count the number of reflexive and symmetric relations on $\{1, 2, \ldots, n\}$ by computing the number of ways to construct the $n \times n$ matrix of a reflexive and symmetric relation.

    The diagonal must consist of all 1's. The $(n^2 - n)/2$ elements above the diagonal can be assigned in $2^{(n^2-n)/2}$ ways. The elements below the diagonal are then determined. Thus there are $2^{(n^2-n)/2}$ reflexive and symmetric relations on an $n$-element set.

81. We count the number of symmetric and antisymmetric relations on $\{1, 2, \ldots, n\}$ by computing the number of ways to construct the $n \times n$ matrix of a symmetric and antisymmetric relation.

    The off-diagonal elements must all be 0's. The $n$ diagonal elements can be assigned in $2^n$ ways. Thus there are $2^n$ symmetric and antisymmetric relations on an $n$-element set.

82. Since the matrix of a reflexive, symmetric, and antisymmetric relation on an $n$-element set must have all diagonal elements 1 and all off-diagonal elements 0, there is one reflexive, symmetric, and antisymmetric relation on an $n$-element set.

84. $n^{n^2}$ 85. $n^{n(n+1)/2}$

87. Let

$$\begin{aligned} X &= \text{eight-bit strings that start with 1} \\ Y &= \text{eight-bit strings that end with 1.} \end{aligned}$$

Then $|X| = |Y| = 2^7$ and $|X \cap Y| = 2^6$. Now $X \cup Y$ is the set of eight-bit strings that start with 1 or end with 1 or both and, by the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 2^7 + 2^7 - 2^6.$$

88. Let

$$\begin{aligned} X &= \text{selections in which Ben is chairperson} \\ Y &= \text{selections in which Alice is secretary.} \end{aligned}$$

Then $|X| = |Y| = 5 \cdot 4$ and $|X \cap Y| = 4$. Now $X \cup Y$ is the set of selections in which Ben is chairperson or Alice is secretary or both and, by the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 20 + 20 - 4 = 36.$$

90. Let

$$\begin{array}{rcl} X & = & \text{selections in which the blue die is 3} \\ Y & = & \text{selections in which the sum is even.} \end{array}$$

Then $|X| = 6$, $|Y| = 18$, and $|X \cap Y| = 3$. Now $X \cup Y$ is the set of selections in which the blue die is 3 or the sum is even or both and, by the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 6 + 18 - 3 = 21.$$

91. Let

$$\begin{array}{rcl} X & = & \text{integers from 1 to 10,000 that are multiples of 5} \\ Y & = & \text{integers from 1 to 10,000 that are multiples of 7.} \end{array}$$

An integer $n$ is a multiple of 5 if there exists an integer $k$ such that $n = 5k$. Thus an integer $n$ from 1 to 10,000 that is a multiple of 5 (i.e., $n = 5k$ for some $k$) satisfies

$$1 \leq 5k \leq 10{,}000.$$

Thus $k$ satisfies

$$\frac{1}{5} \leq k \leq 2000.$$

Since there are 2000 such $k$, $|X| = 2000$. Similarly $|Y| = 1428$. Now $X \cap Y$ consists of integers from 1 to 10,000 that are multiples of both 5 and 7, that is multiples of 35. Arguing as before, we see that $|X \cap Y| = 285$. Also $X \cup Y$ consists of integers from 1 to 10,000 that are multiples of 5 or 7 or both and, by the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 2000 + 1428 - 285 = 3143.$$

92. Letting $A = X$ and $B = Y \cup Z$, we have

$$\begin{array}{rcl} |X \cup Y \cup Z| & = & |X| + |Y \cup Z| - |X \cap (Y \cup Z)| \\ & = & |X| + |Y| + |Z| - |Y \cap Z| - |X \cap (Y \cup Z)| \\ & = & |X| + |Y| + |Z| - |Y \cap Z| - |(X \cap Y) \cup (X \cap Z)| \\ & = & |X| + |Y| + |Z| - |Y \cap Z| - [|X \cap Y| + |X \cap Z| - |X \cap Y \cap Z|] \\ & = & |X| + |Y| + |Z| - |Y \cap Z| - |X \cap Y| - |X \cap Z| + |X \cap Y \cap Z|. \end{array}$$

94. Let $M, P, C$ denote the sets of students taking calculus, psychology, and computer science, respectively. We are given

$$|M \cap P \cap C| = 8,\ |M \cap P| = 20,\ |M \cap C| = 33,\ |P \cap C| = 24,\ |M| = 79,\ |P| = 83,\ |C| = 63.$$

Now

$$\begin{array}{rcl} |M \cup P \cup C| & = & |M| + |P| + |C| - |M \cap P| - |M \cap C| - |P \cap C| + |M \cap P \cap C| \\ & = & 79 + 83 + 63 - 20 - 33 - 24 + 8 = 156. \end{array}$$

Since there are 165 total students, 9 students are taking none of the three subjects.

95. Let $L$ denote the set of persons who watched "Law and Disorder," let $N$ denote the set of persons who watched "25," and let $T$ denote the set of persons who watched "The Tenors." We are given

$$|L| = 68,\ |N| = 61,\ |T| = 52,$$
$$|L \cap N| = 16,\ |L \cap T| = 25,\ |N \cap T| = 19,$$
$$|L \cup N \cup T| = 151 - 26 = 125.$$

Now

$$\begin{aligned} |L \cap N \cap T| &= |L \cup N \cup T| - |L| - |N| - |T| + |L \cap N| + |L \cap T| + |N \cap T| \\ &= 125 - 68 - 61 - 52 + 16 + 25 + 19 = 4. \end{aligned}$$

Thus four persons watched all three shows.

97. Proceed as in the solution given to Exercise 92.

98. $$\begin{aligned} |A \cup B \cup C \cup D| &= |A| + |B| + |C| + |D| \\ &\quad - |A \cap B| - |A \cap C| - |A \cap D| - |B \cap C| - |B \cap D| - |C \cap D| \\ &\quad + |A \cap B \cap C| + |A \cap B \cap D| + |A \cap C \cap D| + |B \cap C \cap D| \\ &\quad - |A \cap B \cap C \cap D| \end{aligned}$$

99. Let $A$ denote the set of integers between 1 and 10,000 that are multiples of 3, let $B$ denote the set of integers between 1 and 10,000 that are multiples of 5, let $C$ denote the set of integers between 1 and 10,000 that are multiples of 11, and let $D$ denote the set of integers between 1 and 10,000 that are multiples of 13. Then

$$|A| = 333,\ |B| = 2000,\ |C| = 909,\ |D| = 769.$$

(The solution to Exercise 91 shows how these numbers were computed.)

Now $A \cap B$ is the set of integers between 1 and 10,000 that are multiples of 3 and 5, that is, multiples of 15. Thus

$$|A \cap B| = 666.$$

Similarly,

$$|A \cap C| = 303,\ |A \cap D| = 256,\ |B \cap C| = 181,\ |B \cap D| = 153,\ |C \cap D| = 69.$$

Now $A \cap B \cap C$ is the set of integers between 1 and 10,000 that are multiples of 3, 5, and 11, that is, multiples of 165. Thus

$$|A \cap B \cap C| = 60.$$

Similarly,

$$|A \cap B \cap D| = 51,\ |A \cap C \cap D| = 23,\ |B \cap C \cap D| = 13.$$

Finally, $A \cap B \cap C \cap D$ is the set of integers between 1 and 10,000 that are multiples of 3, 5, 11, and 13, that is, multiples of 2145. Thus

$$|A \cap B \cap C \cap D| = 4.$$

Thus

$$\begin{aligned} |A \cup B \cup C \cup D| &= |A| + |B| + |C| + |D| \\ &\quad - |A \cap B| - |A \cap C| - |A \cap D| - |B \cap C| - |B \cap D| - |C \cap D| \\ &\quad + |A \cap B \cap C| + |A \cap B \cap D| + |A \cap C \cap D| + |B \cap C \cap D| \\ &\quad - |A \cap B \cap C \cap D| \\ &= 333 + 2000 + 909 + 769 - 666 - 303 - 256 - 181 - 153 - 69 \\ &\quad + 60 + 51 + 23 + 13 - 4 = 2526. \end{aligned}$$

## Section 6.2

2. $abcd, abdc, acbd, acdb, adbc, adcb,$
$bacd, badc, bcad, bcda, bdac, bdca,$
$cabd, cadb, cbad, cbda, cdab, cdba,$
$dabc, dacb, dbac, dbca, dcab, dcba$

3. $P(4,3) = 4(3)(2) = 24$

5. $11!$

6. $P(11,5) = 11 \cdot 10 \cdot 9 \cdot 8 \cdot 7$

8. $P(12,4) = 12 \cdot 11 \cdot 10 \cdot 9$

9. $P(12,3) = 12 \cdot 11 \cdot 10$

11. $3! \cdot 3!$

12. Tokens labeled $AE$, $C$, and $DB$ can be permuted in $3!$ ways.

14. $\frac{1}{2}5!$ since half have $A$ before $D$ and half have $A$ after $D$.

15. We first count the number of strings containing either $AB$ or $CD$ or both. To this end, let

$$\begin{aligned} X &= \text{strings that contain } AB \\ Y &= \text{strings that contain } CD. \end{aligned}$$

By previous methods, we find that

$$|X| = |Y| = 4!, \quad |X \cap Y| = 3!.$$

Now $X \cup Y$ is the set of strings that contain either $AB$ or $CD$ or both. By the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 4! + 4! - 3!.$$

Since there are $5!$ total strings, the number that contain neither $AB$ nor $CD$ is $5! - (4! + 4! - 3!)$.

17. $C(5,3) \cdot 2!$. Pick three slots for $A$, $C$, and $E$. Then place the two remaining letters.

18. Let

$$\begin{aligned} X &= \text{strings that contain } DB \\ Y &= \text{strings that contain } BE. \end{aligned}$$

By previous methods, we find that

$$|X| = |Y| = 4!, \quad |X \cap Y| = 3!.$$

Now $X \cup Y$ is the set of strings that contain $DB$ or $BE$ or both. By the Inclusion-Exclusion Principle,

$$|X \cup Y| = |X| + |Y| - |X \cap Y| = 4! + 4! - 3!.$$

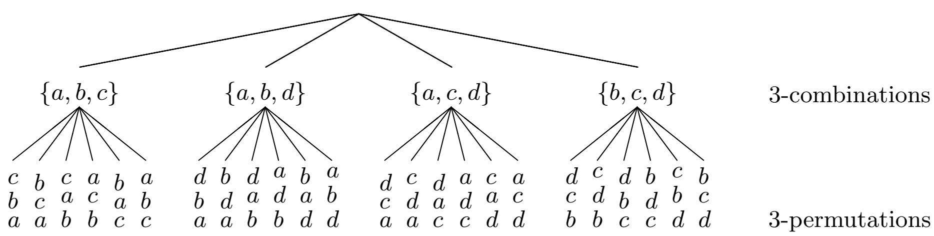
20. First, line up the Vesuvians and the Jovians. This can be done in 18! ways. For each of these arrangements, we can place the Martians in 5 of the 19 in-between and end positions, which can be done in $P(19, 5)$ ways. Thus there are $18! \cdot P(19, 5)$ arrangements.

22. 9!

23. Seat the Martians (4! ways). Seat the Jovians in the in-between spots ($5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ ways). The answer is $4! \cdot 5!$.

26. $\{a, b, c\}, \{a, b, d\}, \{a, c, d\}, \{b, c, d\}$

27.




29. $C(12, 4)$

30. $C(44, 6), \quad C(48, 6)$

32. The solution to Exercise 31 shows that there are 838 different pizzas. In selecting four pizzas, there are four possibilities:

(a) Four kinds of pizzas.
(b) Three kinds of pizzas.
(c) Two kinds of pizzas.
(d) One kind of pizza.

[Possibility (a).] Four kinds of pizzas can occur in $C(838, 4)$ ways.

[Possibility (b).] Three distinct of pizzas can occur in $C(838, 3)$ ways, and then the fourth pizza is the same as one of the three. Thus three kinds of pizzas can occur in $3C(838, 3)$ ways.

[Possibility (c).] Two distinct pizzas can occur in $C(838, 2)$ ways. The two remaining pizzas can occur in three ways: the two can be the same as the first pizza, the two can be the same as the second pizza, one can be the same as the first and the other the same as the second. Thus two kinds of pizzas can occur in $3C(838, 2)$ ways.

[Possibility (d).] One kind of pizza can occur in $C(838, 1)$ ways.

Thus the total number of ways that four pizzas can be selected is

$$C(838, 4) + 3C(838, 3) + 3C(838, 2) + C(838, 1).$$

34. $C(6,3)C(7,4)$

35. $C(13,4) - C(6,4)$ [The number of possible committees is $C(13,4)$. The number that have no women is $C(6,4)$.]

37. $C(13,4) - [C(6,4) + C(7,4)]$ (The total number of minus the number with all men or all women.)

38. $C(13,4) - C(11,2)$ [The number of possible committees is $C(13,4)$. The number in which Mabel and Ralph serve together is $C(11,2)$.]

40. $C(8,3)$

41. Six: 00011111, 10001111, 11000111, 11100011, 11110001, 11111000

44. $13 \cdot C(48,1)$ (Choose the denomination and then the odd card.)

45. $C(13,5)$

47. $4 \cdot C(13,2) \cdot 13^3$ (You must pick two of one suit and one of each of the three remaining suits. First choose the suit to have two cards. Then choose two cards. Then choose one of each of the remaining suits.)

48. 4

50. $9 \cdot 4^5$ (Pick the lowest card's denomination. Then pick the suit of each of the denominations.)

51. $C(13,2)C(11,1)C(4,2)C(4,2)C(4,1)$ (Pick the two denominations that receive two cards; pick the denomination to receive one card; pick two cards from each of the chosen denominations; pick one card from the other denomination.)

53. 4

54. $C(4,2)[C(26,13) - 2]$ [Pick the two suits. The number of hands containing cards from the chosen suits is $C(26,13)$. Subtract the two hands that contain only cards of one of the chosen suits.]

56. $C(13,5)C(13,4)C(13,3)C(13,1)$

57. $4!C(13,5)C(13,4)C(13,3)C(13,1)$ [Pick the suits (the order determines which gets $5,4,3,1$). Then pick the desired number of cards from the selected suits.]

59. $C(32,13)$ (Select 13 cards from among the 32 non-face cards.)

61. $C(10,3)$

62. $C(10,3) + C(10,2) + C(10,1) + C(10,0)$

64. $C(10,5)$

66. $C(46,4)$

67. $C(46,2)C(4,2)$ (Select 2 good and 2 defective.)

69. Represent each of the two 10's by a star ("$*$"). The remaining $n-4$ bits can be placed in the three in-between and end positions with respect to the two stars. For each of these three groups of bits, if a 1 occurs, the bits to its right in that group must also be 1's. We place a vertical bar "|" in between the 0's and 1's in each of these groups. Each string can be represented by $0^a|1^b * 0^c|1^d * 0^e|1^f$, where $0 \le a,b,c,d,e,f \le n-4$ and $a+b+c+d+e+f = n-4$. Note that the length of the string is $n+1$. A particular string is determined by the choice of five slots from $n+1$ slots for the pattern $| * | * |$. Hence there are $C(n+1,5)$ such strings.

70. Fix $n-k$ 1's. The $k$ 0's must be assigned to the $n-k+1$ positions between the 1's or at either end. This can be done in $C(n-k+1,k)$ ways.

71. Look at the formula for $C(n,k)$.

74. Argue as in Example 6.2.23.

75. Note that the minimum number of votes for Wright is $\lceil n/2 \rceil$. Thus, by Exercise 74, the number of ways the votes could be counted is

$$1 + \sum_{r=\lceil n/2 \rceil}^{n-1} [C(n,r) - C(n,r+1)] = C(n, \lceil n/2 \rceil).$$

(The first term, 1, is the number of ways Wright receives $n$ votes and Upshaw receives 0 votes.)

77. By Exercise 75, $k$ vertical steps can occur in $C(k, \lceil k/2 \rceil)$ ways, since, at any point, the number of up steps is greater than or equal to the number of down steps. Then, $n-k$ horizontal steps can be inserted among the $k$ vertical steps in $C(n,k)$ ways. These $n-k$ horizontal steps can occur in $C(n-k, \lceil (n-k)/2 \rceil)$ ways, since, at any point, the number of right steps is greater than or equal to the number of left steps. Thus, the number of paths that stay in the first quadrant containing exactly $k$ vertical steps is

$$C(k, \lceil k/2 \rceil) C(n,k) C(n-k, \lceil (n-k)/2 \rceil).$$

Summing over all $k$, we find that the total number of paths is

$$\sum_{k=0}^{n} C(k, \lceil k/2 \rceil) C(n,k) C(n-k, \lceil (n-k)/2 \rceil).$$

78. Fix a starting position, and move around the table. When a handshake begins, write $R$; when a handshake ends, write a $U$. The result is a sequence of $n$ $R$'s and $n$ $U$'s in which the number of $R$'s is always greater than or equal to the number of $U$'s. Furthermore, the correspondence between such sequences and handshakes is one-to-one and onto. Since the number of sequences of $n$ $R$'s and $n$ $U$'s in which the number of $R$'s is always greater than or equal to the number of $U$'s is $C_n$ (see Example 6.2.23), the number of ways that $2n$ persons seated around a circular table can shake hands in pairs without any arms crossing is also $C_n$.

79. We show a one-to-one, onto correspondence between the output $i_1, i_2, \ldots, i_n$ reversed and sequences of $n$ $R$'s and $n$ $U$'s in which the number of $R$'s is always greater than or equal to the number of $U$'s. Since the number of such sequences is $C_n$, the result then follows.

   For each sequence of $n$ $R$'s and $n$ $U$'s in which the number of $R$'s is always greater than or equal to the number of $U$'s, under each $R$ write the number of $D$'s that precede it. For example, for $n=3$ we would have

$$\begin{array}{cccccc} R & R & D & R & D & D \\ 0 & 0 & & 1 & & \end{array}$$

   Then add one to each value of the resulting sequence. In our case, we obtain the sequence 112. For $n=3$, the complete correspondence is

| *RD Sequence* | *Numeric Sequence* | *Numeric Sequence Reversed (Output)* |
|---|---|---|
| $RRRDDD$ | 111 | 111 |
| $RRDRDD$ | 112 | 211 |
| $RRDDRD$ | 113 | 311 |
| $RDRRDD$ | 122 | 221 |
| $RDRDRD$ | 123 | 321 |

80. There are $P(n, r)$ ways to place the $r$ distinct objects. There is one way to place the identical objects in the remaining slots. Thus the total number of orderings is $P(n, r)$.

    There are $C(n, n - r) = C(n, r)$ ways to choose positions for the $n - r$ identical objects. After placing the identical objects in these positions, there are $r!$ ways to place the distinct objects. Thus the number of total orderings is $r!C(n, r)$. Therefore

$$P(n, r) = r!C(n, r).$$

81. The Addition Principle must be applied to a family of pairwise disjoint sets. Here the sets involved are *not* pairwise disjoint. For example, if $X$ is the set of hands containing clubs, diamonds, and spades and $Y$ is the set of hands containing clubs, diamonds, and hearts, $X \cap Y$ contains hands that contain clubs and diamonds.

83. Double-counting occurs. For a specific example, consider

$$X = \{1, 2, 3, 4\}, \quad Y = \{a, b, c\}.$$

    We first select a 3-permutation of $X$, say $2, 1, 4$. We then assign 2 the value $a$, 1 the value $b$, and 4 the value $c$. We then assign the remaining element of $X$, namely 3, an arbitrary value in $Y$: Suppose that we assign 3 the value $b$. The function is

$$\{(1, b), (2, a), (3, b), (4, c)\}.$$

    Now suppose that we select the 3-permutation $2, 3, 4$ of $X$. We then assign 2 the value $a$, 3 the value $b$, and 4 the value $c$. We then assign the remaining element of $X$, namely 1, an arbitrary value in $Y$: Suppose that we assign 1 the value $b$. The function is

$$\{(1, b), (2, a), (3, b), (4, c)\},$$

    which is the same as the first function.

85. Six, since there are six ways to choose five of the slots marked 0, after which we can fill the remaining slots with 0's and 1's to obtain the string 10001000.

86. $C(8, 5)$, since there are $C(8, 5)$ way to choose five slots to fill with 0's, after which we can fill the remaining slots with 0's to obtain the string 00000000.

87. (a) If there are more tables than people, it is impossible to seat at least one person at each table.

    (b) If there are equal numbers of tables and people and there is at least one person at each table, there will be exactly one person at each table.

(c) See Example 6.2.7.

(d) In this case, there are two people at one table and one person at each other table. The two people to sit at one table can be chosen in $C(n, 2)$ ways.

(e) We prove this equation by induction on $n$. The Basis Step, $n = 2$, is part (b).

Assume that the equation is true for $n$ and that we have $n+1$ people. Choose one person. Either this person sits alone or with others. If the person sits alone, the other persons can be seated at the other table in $(n-1)!$ ways. If this person is not alone, by the inductive hypothesis the remaining $n$ persons can be seated in

$$(n-1)! \sum_{i=1}^{n-1} \frac{1}{i}$$

ways. The $(n+1)$st person can be added to this seating arrangement in $n$ ways (to the right of any of the other $n$ persons). Thus there are

$$n \cdot (n-1)! \sum_{i=1}^{n-1} \frac{1}{i} = n! \sum_{i=1}^{n-1} \frac{1}{i}$$

ways to seat $n+1$ people if the $(n+1)$st person does not sit alone. Therefore the total number of seatings is

$$(n-1)! + n! \sum_{i=1}^{n-1} \frac{1}{i} = n! \sum_{i=1}^{n} \frac{1}{i}.$$

The Inductive Step is complete.

(f) Fix $n$. Each seating of $n$ persons at $k$ round tables, with at least one person at each table, determines a unique permutation of $1, \ldots, n$. If $p(i, 1), \ldots, p(i, e_i)$ are seated clockwise at table $i$, $i = 1, \ldots, k$, in this order, we interpret this as the permutation defined by the mapping

$$\begin{array}{rcl}
p(1,1) & \to & p(1,2) \\
p(1,2) & \to & p(1,3) \\
 & \vdots & \\
p(1,e_1-1) & \to & p(1,e_1) \\
p(1,e_1) & \to & p(1,1) \\
 & \vdots & \\
p(k,1) & \to & p(k,2) \\
p(k,2) & \to & p(k,3) \\
 & \vdots & \\
p(k,e_k-1) & \to & p(k,e_k) \\
p(k,e_k) & \to & p(k,1)
\end{array}$$

(This representation is called the *decomposition of a permutation into its cycles.*) Since all permutations are accounted for, the equation follows.

(g) We show that $s_{3,1} = 2$ and

$$s_{n,n-2} = 2C(n, n-3) + 3C(n, n-4)$$

for $n \geq 4$.

$s_{3,1} = 2$ by part (c).

If $n \geq 4$, there are two basic seating arrangements:

1. $n-3$ tables of one person each and one table of three persons. There are $2C(n, n-3)$ such seatings since we may select the $n-3$ solitary persons in $C(n, n-3)$ ways, and then seat the remaining three persons at one table in 2! ways (using the formula from part (c)).
2. $n-4$ tables of one person each and two tables of two persons each. Select the $n-4$ solitary persons in $C(n, n-4)$ ways, and then seat the remaining four persons in three ways. In this case, there are $3C(n, n-4)$ seatings.

88. (a) If $k > n$, an $n$-element set cannot be partitioned into $k$ nonempty subsets.

(b) There is one way to partition an $n$-element set into $n$ nonempty subsets: Each subset must consist of one element.

(c) There is one way to partition an $n$-element set into one nonempty subset: The subset is the $n$-element set itself.

(d)–(f) See (g) and (h).

(g) Let $X$ be an $n$-element set and let $x \in X$. For each nonempty subset $Y$ of $X - \{x\}$, $\{Y, X - Y\}$ is a partition of $X$. Since these are also all the partitions, $S_{n,2} = 2^{n-1} - 1$.

(h) A partition of an $n$-element set into $n-1$ subsets consists of a subset containing two elements and $n-2$ subsets each containing one element. The 2-element subset can be chosen in $C(n, 2)$ ways. Therefore $S_{n,n-1} = C(n, 2)$.

(i) $S_{n,n-2} = C(n, 3) + 3C(n, 4)$.

If we partition an $n$-element set into $n-2$ subsets, either there is a subset consisting of three elements with all other subsets consisting of one element [there are $C(n, 3)$ of these], or there are two subsets each consisting of two elements with all other subsets consisting of one element [there are $C(n, 4)$ ways to choose the elements to be the doubletons and three ways to organize the four elements into doubletons].

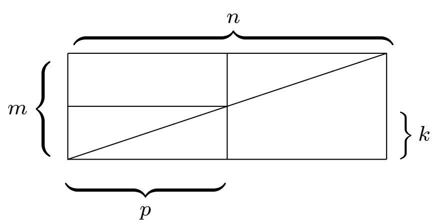
90. $P(m, n)$, since each $n$-permutation of $Y$ determines a one-to-one function from $X$ to $Y$ and conversely. If $X = \{x_1, \ldots, x_n\}$ and $y_1, \ldots, y_n$ is an $n$-permutation of $Y$, $f(x_i) = y_i$ is a one-to-one function from $X$ to $Y$. Similarly, if $f$ is a one-to-one function from $X$ to $Y$, $f(x_1), \ldots, f(x_n)$ is an $n$-permutation of $Y$.

91. $n!$. The solution to Exercise 90 show that there are $P(m, n)$ one-to-one functions from an $n$-element set to an $m$-element set. Taking $m = n$, we find that there are $P(n, n) = n!$ one-to-one functions from an $n$-element set to an $n$-element set.

## Problem-Solving Corner: Combinations

1. From the following figure, we derive the formula

$$\sum_{k=0}^{m} C(k+p,p)C(m-k+n-p,n-p) = C(m+n,m)$$



2. $\sum_{k=0}^{\min\{m,n\}} C(m,k)C(n,k) = C(m+n,m)$

## Section 6.3

2. $6!/2!$ 3. $12!/(4!2!)$

5. We form strings in which no two $S$'s are consecutive by first placing the letters *ALEPERON*, which can be done in

$$\frac{8!}{2!}$$

ways. We then place the four $S$'s in the nine in-between positions

$$\_\_\ A\ \_\_\ L\ \_\_\ E\ \_\_\ P\ \_\_\ E\ \_\_\ R\ \_\_\ O\ \_\_\ N\ \_\_,$$

which can be done in $C(9,4)$ ways. Thus

$$\frac{C(9,4)8!}{2!}$$

strings can be formed by ordering the letters *SALESPERSONS* if no two $S$'s are consecutive.

6. We count the number of strings of length zero, the number of strings of length one, and so on, and then sum these numbers. The number of strings of length zero is one, and the number of strings of length one is five.

There is one string of length two that uses the two $O$'s, and there are $5 \cdot 4$ strings of length two that do not use two $O$'s (formed by selecting a 2-permutation of *SCHOL*). Thus there are 21 strings of length two.

There are four ways to choose three letters including two $O$'s. There are $\frac{3!}{2!} = 3$ ways to permute these letters. Thus there are $4 \cdot 3$ strings of length three that use the two $O$'s, and there are $5 \cdot 4 \cdot 3$ strings of length three that do not use two $O$'s (formed by selecting a 3-permutation of *SCHOL*). Thus there are 72 strings of length three.

There are $C(4,2) = 6$ ways to choose four letters including two $O$'s. There are $\frac{4!}{2!} = 6$ ways to permute these letters. Thus there are $6 \cdot 6 = 36$ strings of length four that use the two $O$'s, and there are $5 \cdot 4 \cdot 3 \cdot 2$ strings of length four that do not use two $O$'s (formed by selecting a 4-permutation of *SCHOL*). Thus there are 156 strings of length four.

There are $C(4,3) = 4$ ways to choose five letters including two $O$'s. There are $\frac{5!}{2!} = 60$ ways to permute these letters. Thus there are $4 \cdot 60 = 240$ strings of length five that use the two $O$'s, and there are $5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$ strings of length four that do not use two $O$'s (formed by selecting a 5-permutation of *SCHOL*). Thus there are 360 strings of length five.

There are $\frac{6!}{2!} = 360$ strings of length six. Thus there are

$$1 + 5 + 21 + 72 + 156 + 360 + 360 = 975$$

strings that can be formed by ordering the letters *SCHOOL* using some or all of the letters.

8. $C(10 + 6 - 1, 6 - 1)$ 9. $C(4 + 6 - 1, 6 - 1)$

11. Assign each problem five points, and let $x_i$ denote the number of additional points that can be assigned to problem $i$. Now the question is: How many solutions are there of

$$\sum_{i=1}^{12} x_i = 40?$$

Arguing as in Example 6.3.8, the answer is $C(40 + 12 - 1, 12 - 1)$.

12. $4^{100}$ 13. $C(100 + 4 - 1, 4 - 1)$ 16. $C(9 + 3 - 1, 9)$ 17. $C(4 + 3 - 1, 4)$

19. $C(8 + 2 - 1, 8)$ 20. $C(10 + 2 - 1, 10) + C(9 + 2 - 1, 9)$ 23. $C(12 + 3 - 1, 12)$

24. $C(14 + 2 - 1, 14)$ 26. $C(15 + 3 - 1, 15) - C(8 + 3 - 1, 8)$

27. There are $C(14+3-1, 14)$ solutions satisfying $0 \leq x_1$, $1 \leq x_2$, $0 \leq x_3$. Of these, $C(8+3-1, 8)$ have $x_1 \geq 6$; $C(6 + 3 - 1, 6)$ have $x_2 \geq 9$; and there is one with $x_1 \geq 6$ and $x_2 \geq 9$. Thus there are

$$C(8 + 3 - 1, 8) + C(6 + 3 - 1, 6) - 1$$

solutions with $x_1 \geq 6$ or $x_2 \geq 9$. Therefore there are

$$C(14 + 3 - 1, 14) - [C(8 + 3 - 1, 8) + C(6 + 3 - 1, 6) - 1]$$

of the desired type.

29. The number of solutions to the given equation, where $x_1$, $x_2$, and $x_3$ are positive integers, is the same as the number of solutions to

$$x_1 + x_2 + x_3 = n - 3,$$

where $x_1$, $x_2$, and $x_3$ are nonnegative integers (just add 1 to each of $x_1$, $x_2$, and $x_3$). The number of solutions to the latter equation is

$$C(n - 3 + 3 - 1, n - 3) = C(n - 1, n - 3) = \frac{(n - 1)(n - 2)}{2}.$$

30. The problem is equivalent to solving

$$x_1 + x_2 + \cdots + x_n + x_{n+1} = M$$

since

$$0 \leq x_{n+1} = M - x_1 + x_2 + \cdots + x_n.$$

Thus the number of solutions is

$$C(M + (n+1) - 1, (n+1) - 1) = C(M + n, n).$$

31. We must count the number of solutions of

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 15$$

satisfying $0 \leq x_i \leq 9$, $i = 1, \ldots, 6$. There are $C(15+6-1, 15)$ solutions with $x_i \geq 0$, $i = 1, \ldots, 6$. There are $C(5 + 6 - 1, 5)$ with $x_1 \geq 10$. There are $6C(5 + 6 - 1, 5)$ with some $x_i \geq 10$. (Note that there is no double counting, since we cannot have $x_i \geq 10$ and $x_j \geq 10$, $i \neq j$.) Thus the solution is $C(15 + 6 - 1, 15) - 6C(5 + 6 - 1, 5)$.

32. $C(20 + 6 - 1, 20) - [6C(10 + 6 - 1, 10) - C(6, 2)]$

34. $8!/(4!{\cdot}2!{\cdot}2!)$

35. $C(7{+}2{-}1, 2)$

37. $C(5{+}3{-}1, 5)$

38. $C(6, 2)C(6, 3)C(8, 2)$

40. $C(20, 5)C(15, 5)$

41. $[C(20, 5) - C(14, 5)][C(14, 5) + 6C(14, 4)]$

43. $C(15 + 5 - 1, 15)C(10 + 5 - 1, 10)$

44. $C(12, 10)$

46. Consider the number of orderings of $kn$ objects where there are $n$ identical objects of each of $k$ types.

47. The number of times the print statement is executed is

$$1 + 2 + \cdots + n.$$

Example 6.3.9 shows that this is the same as $C(2 + n - 1, 2) = (n + 1)n/2$.

49. *list_sols*$(n)$ {
    for $x_1 = 0$ to $n$
        for $x_2 = 0$ to $n - x_1$
            *println*$(x_1, x_2, n - x_1 - x_2)$
}

50. Many partitions are not counted. For example, the partition

$$\{\{x_1, x_3\}, \{x_2\}, \{x_4\}, \{x_5\}, \{x_6\}, \{x_7\}, \{x_8\}, \{x_9, x_{10}\}\}$$

is not counted.

## Section 6.4

2. 12456 3. 23456 5. 631245 6. 13245678

8. (For Exercise 5) After the while loop in lines 7–9 finishes, $m$ is 2. After the while loop in lines 11–13 finishes, $k$ is 5. At line 14, we swap $s_2$ and $s_5$. Now the sequence is 635421. The while loop of lines 17–22 reverses $s_3, \ldots, s_6$. The result is 631245.

10. 12, 13, 14, 15, 16, 23, 24, 25, 26, 34, 35, 36, 45, 46, 56

11. 12345, 12346, 12347, 12356, 12357, 12367, 12456, 12457, 12467, 12567, 13456, 13457, 13467, 13567, 14567, 23456, 23457, 23467, 23567, 24567, 34567

13. 123, 132, 213, 231, 312, 321

15. Change line 5 to

    while (true) {

    Add the following lines after line 13 to the body of the while loop at lines 11–13

    if ($k == 0$)
      return

    (Also, since there will now be multiple lines in the body of the while loop, enclose them in braces.)

16. Input: $r, n$
    Output: A list of all $r$-permutations of $\{1, 2, ..., n\}$

```
list_r_perms(r, n) {
   for i = 1 to r
      s_i = i
   r_comb(r)
   for i = 2 to C(n, r) {
      find the rightmost s_m not at its maximum value
      s_m = s_m + 1
      for j = m + 1 to r
         s_j = s_j + 1
      r_comb(r)
   }
}

r_comb(r) {
   for i = 1 to r
      t_i = s_i
   println(t)
   for i = 2 to r! {
      find the largest index m satisfying t_m < t_{m+1}
```

```
        find the largest index k satisfying t_k > t_m
        swap(t_m, t_k)
        reverse the order of the elements t_{m+1}, ..., t_r
        println(t)
    }
}
```

18. Input: $s_1, \ldots, s_n$ (a permutation of $\{1, \ldots, n\}$) and $n$
Output: $s_1, \ldots, s_n$, the next permutation. (The first permutation follows the last permutation.)

```
next_perm(s, n) {
    s_0 = 0 // dummy value
    m = n − 1
    while (s_m > s_{m+1})
        m = m − 1
    k = n
    while (s_m > s_k)
        k = k − 1
    if (m > 0)
        swap(s_m, s_k)
    p = m + 1
    q = n
    while (p < q) {
        swap(s_p, s_q)
        p = p + 1
        q = q − 1
    }
}
```

20. Input: $s_1, \ldots, s_n$ (a permutation of $\{1, \ldots, n\}$) and $n$
Output: $s_1, \ldots, s_n$, the previous permutation. (The last permutation precedes the first permutation.)

```
prev_perm(s, n) {
    s_0 = n + 1 // dummy value
    // working from right, find first index where s_i > s_{i+1}
    i = n − 1
    while (s_i < s_{i+1})
        i = i − 1
    // reverse s_{i+1}, ..., s_n
    j = i + 1
    k = n
    while (j < k) {
        swap(s_j, s_k)
```

$j = j + 1$
$k = k - 1$
}
// if $i > 0$, swap $s_i$ with the first $s$ value after $s_i$ that is less than $s_i$
if $(i > 0)$ {
$j = i + 1$
while $(s_j > s_i)$
$j = j + 1$
$swap(s_i, s_j)$
}
}

22. Input: $s_1, \ldots, s_n, n$, and a string $\alpha$
Output: All permutations of $s_1, \ldots, s_n$, each prefixed by $\alpha$. (To list all permutations of $s_1, \ldots, s_n$, invoke this procedure with $\alpha$ equal to the null string.)

$perm\_recurs(s, n, \alpha)$ {
if $(n == 1)$ {
$println(\alpha + s_1)$
return
}
for $i = 1$ to $n$ {
$\alpha' = \alpha + s_i$
$perm\_recurs(\{s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n\}, n - 1, \alpha')$
}
}

## Section 6.5

2. (H,2), (H,4), (H,6)
3. (H,1), (H,3), (H,5), (T,1), (T,3), (T,5)
6. (1,1), (2,2), (3,3), (4,4), (5,5), (6,6)
7. (1,4), (2,4), (3,4), (4,4), (5,4), (6,4), (4,1), (4,2), (4,3), (4,5), (4,6)
9. Suppose that the experiment is: Roll three dice. An event is: The sum of the dice is 8.
10. Suppose that the experiment is: Roll three dice. The sample space is the set of all possible outcomes. (There are 216 possible outcomes.)
12. $\frac{3}{6}$
13. $\frac{5}{6}$
15. $\frac{4}{52}$
16. $\frac{13}{52}$
18. Since an odd sum can be obtained in 18 ways,

$$\begin{array}{l}(1,2),\ (1,4),\ (1,6),\ (2,1),\ (2,3),\ (2,5),\ (3,2),\ (3,4),\ (3,6),\\ (4,1),\ (4,3),\ (4,5),\ (5,2),\ (5,4),\ (5,6),\ (6,1),\ (6,3),\ (6,5),\end{array}$$

the probability of obtaining an odd sum is $\frac{18}{36}$.

19. Since doubles can be obtained in six ways, (1,1), (2,2), (3,3), (4,4), (5,5), (6,6), the probability of obtaining doubles is $\frac{6}{36}$.

21. Exactly one defective microprocessor can be obtained in $10 \cdot C(90, 3)$ ways. (Choose one defective microprocessor and choose three good microprocessors.) Since four microprocessors can be chosen in $C(100, 4)$ ways, the probability of obtaining exactly one defective microprocessor is

$$\frac{10 \cdot C(90,3)}{C(100,4)}.$$

22. At most one defective microprocessor can be obtained in $C(90, 4) + 10 \cdot C(90, 3)$ ways. (Choose four good microprocessors, or choose one defective microprocessor and three good microprocessors.) Since four microprocessors can be chosen in $C(100, 4)$ ways, the probability of obtaining at most one defective microprocessor is

$$\frac{C(90,4) + 10 \cdot C(90,3)}{C(100,4)}.$$

24. $\dfrac{3!}{10^3}$ 25. $\dfrac{1}{C(49,6)}$ 27. $\dfrac{1}{C(31,7)}$ 29. $\dfrac{C(26,13)}{C(52,13)}$ 31. $\dfrac{1}{2^{10}}$

32. $\dfrac{10}{2^{10}}$ 35. $1 - \dfrac{1}{3^{10}}$ 36. $\dfrac{1}{3^{10}}$

39. An equivalent problem is to count strings of three $C$'s and nine $N$'s in which no two $C$'s are consecutive since we can regard the positions of the $C$'s as representing the chosen lockers. For example, the string $NNCNNNCNNNNC$ represents the choice of lockers 3, 7, and 12, no two of which are consecutive. We can obtain such strings by placing the three $C$'s in the 10 in-between positions of the nine $N$'s:

$$\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_\, N \,\_$$

The number of ways of choosing the positions for the $C$'s is, thus, $C(10, 3)$. Therefore the probability that no two lockers are consecutive is

$$\frac{C(10,3)}{C(12,3)}.$$

40. $1 - \dfrac{C(10,3)}{C(12,3)}$ 42. $\left(\dfrac{18}{38}\right)^2$ 43. $\frac{1}{38}$

46. If you make a random decision, you are choosing randomly between two doors—one with a goat and one with the car; therefore, the probability of winning the car is $\frac{1}{2}$.

47. Suppose that behind the first two doors are goats, and behind the third door is the car. Consider your three initial choices. If you initially choose door one, your switch will move you to the door with the car, and you win. Similarly, if you initially choose door two, your switch will move you to the door with the car, and you win. However, if you initially choose door three, your switch will move you to a door with a goat, and you lose. Therefore the probability of winning the car is $\frac{2}{3}$.

48. One-third of the time the host picks a door that hides a car and the game is over. Consider the two-thirds of the time when the host picks a door that hides a goat. Either the contestant has picked a door with a car behind it or a door with a goat behind it, so the contestant wins one-third of the time regardless of which strategy was followed.

50. If you make a random decision, you are choosing randomly between three doors—two with goats and one with the car; therefore, the probability of winning the car is $\frac{1}{3}$.

51. Suppose that behind the first three doors are goats, and behind the fourth door is the car. There are eight equally probable outcomes. If you initially choose door one, you switch to either a door with a goat or a door with a car. Similarly, if you initially choose door two or three, you switch to either a door with a goat or a door with a car. If you initially choose door four, you switch to one of two doors, each of which hides a goat. Of the eight possibilities, three win a car. Therefore the probability of winning the car is $\frac{3}{8}$.

53. The reasoning is not correct. As a small example, suppose that the sample space consist of eight eggs:
$$g_1, g_2, g_3, g_4, g_5, g_6, b_1, b_2,$$
where $g_i$ denotes a good egg and $b_i$ denotes a bad egg. Then the probability of a bad egg is $1/4$. However, the set $\{g_1, g_2, g_3, g_4\}$ of four eggs contains no bad eggs.

54. There are four ways that the sequence of tosses can start: TT, HT, TH, HH. If the sequence starts TT, the second player wins. If the sequence starts HT, the first player wins. If the sequence begins TH or HH, the first T after the first two coin tosses must be preceded by H and the first player wins. Thus the probability that the first player wins is $3/4$. It's best to be the first player!

56. The 10 disks can be given to Mary, Ivan, and Juan in $C(10+3-1, 3-1)$ ways. If Ivan receives exactly three disks, the remaining seven disks can be given to Mary and Juan in $C(7+2-1, 2-1)$ ways. Thus the probability that Ivan receives exactly three disks is
$$\frac{C(7+2-1, 2-1)}{C(10+3-1, 3-1)}.$$

## Section 6.6

2. Since
$$P(2) = P(4) = P(6) = \frac{1}{8},$$
the probability of getting an even number is
$$P(2) + P(4) + P(6) = \frac{3}{8}.$$

3. Since the probability of getting a 5 is $\frac{1}{8}$, the probability of not getting a 5 is $1 - \frac{1}{8} = \frac{7}{8}$.

5. $\frac{1}{4}$

6. $\frac{1}{4}$

9. The sum of 7 is obtained in six ways: (1,6), (2,5), (3,4), (4,3), (5,2), (6,1). Now

$$P(1,6) = P(1)P(6) = \frac{1}{4} \cdot \frac{1}{12} = \frac{1}{48}.$$

Similarly,

$$P(2,5) = P(3,4) = P(4,3) = P(5,2) = P(6,1) = \frac{1}{48}.$$

Therefore the probability of the sum of 7 is

$$6\left(\frac{1}{48}\right) = \frac{1}{8}.$$

10. Doubles or a sum of 6 is obtained in 10 ways: (1,5), (2,4), (3,3), (4,2), (5,1), (1,1), (2,2), (4,4), (5,5), (6,6). Now

$$P(1,5) = P(1)P(5) = \left(\frac{1}{4}\right)^2 = \frac{1}{16}.$$

Similarly,

$$P(3,3) = P(5,1) = P(1,1) = P(5,5) = \frac{1}{16},$$

$$P(2,4) = P(4,2) = P(2,2) = P(4,4) = P(6,6) = \frac{1}{144}.$$

Therefore the probability of getting doubles or the sum of 6 is

$$5\left(\frac{1}{16} + \frac{1}{144}\right).$$

12. Let $E_1$ be the event "sum of 6," let $E_2$ be the event "doubles," and let $E_3$ be the event "at least one 2." We want

$$P(E_1 \cup E_2 \mid E_3) = \frac{P((E_1 \cup E_2) \cap E_3)}{P(E_3)}.$$

The event $(E_1 \cup E_2) \cap E_3$ comprises (2,4), (4,2), (2,2); therefore,

$$P((E_1 \cup E_2) \cap E_3) = 3\left(\frac{1}{12}\right)^2.$$

The solution to Exercise 11 shows that

$$P(E_3) = \frac{23}{144}.$$

Therefore

$$P(E_1 \cup E_2 \mid E_3) = \frac{P((E_1 \cup E_2) \cap E_3)}{P(E_3)} = \frac{\frac{3}{144}}{\frac{23}{144}} = \frac{3}{23}.$$

13. Let $E_1$ be the event "sum of 6," let $E_2$ be the event "sum of 8," and let $E_3$ be the event "at least one 2." We want

$$P(E_1 \cup E_2 \mid E_3) = \frac{P((E_1 \cup E_2) \cap E_3)}{P(E_3)}.$$

The event $(E_1 \cup E_2) \cap E_3$ comprises (2,4), (4,2), (2,6), (6,2); therefore,

$$P((E_1 \cup E_2) \cap E_3) = 4\left(\frac{1}{12}\right)^2.$$

The solution to Exercise 11 shows that

$$P(E_3) = \frac{23}{144}.$$

Therefore

$$P(E_1 \cup E_2 \mid E_3) = \frac{P((E_1 \cup E_2) \cap E_3)}{P(E_3)} = \frac{\frac{4}{144}}{\frac{23}{144}} = \frac{4}{23}.$$

15. (H,3) 16. No 18. No 20. $1 - \dfrac{C(90,10)}{C(100,10)}$

21. $\dfrac{C(10,3)C(90,3) + C(10,4)C(90,2) + C(10,5)C(90,1) + C(10,6)}{C(100,10)}$

23. $\dfrac{C(4,2)}{2^4}$

24. The probability of all boys or all girls is $\frac{2}{2^4}$, so the probability of at least one boy and at least one girl is

$$1 - \frac{2}{2^4}.$$

26. Let $E$ be the event "exactly two girls," and let $F$ be the event "at least one girl." We want to compute

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)}.$$

Now

$$P(E \cap F) = P(E) = \frac{C(4,2)}{2^4},$$

and

$$P(F) = 1 - P(\text{no girls}) = 1 - \frac{1}{2^4}.$$

Therefore

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)} = \frac{\frac{C(4,2)}{2^4}}{1 - \frac{1}{2^4}}.$$

27. Let $E$ be the event "at least one boy," and let $F$ be the event "at least one girl." We want to compute

$$P(E \cap F \mid F) = \frac{P((E \cap F) \cap F)}{P(F)} = \frac{P(E \cap F)}{P(F)}.$$

Now

$$P(E \cap F) = 1 - \frac{2}{2^4},$$

and

$$P(F) = 1 - \frac{1}{2^4}.$$

Therefore

$$P(E \cap F \,|\, F) = \frac{P(E \cap F)}{P(F)} = \frac{1 - \frac{2}{2^4}}{1 - \frac{1}{2^4}}.$$

29. Let $E$ be the event "at most one boy," and let $F$ be the event "at most one girl." Then $P(E) = P(F) = 1 - \frac{5}{2^4}$, and $P(E \cap F) = 0$. Since

$$P(E \cap F) = 0 \neq \left(1 - \frac{5}{2^4}\right)^2 = P(E)P(F),$$

$E$ and $F$ are not independent.

30. Let $E$ be the event "children of both sexes," and let $F$ be the event "at most one girl." Then

$$P(E) = 1 - \frac{2}{2^n}, \quad P(F) = \frac{n+1}{2^n}, \quad P(E \cap F) = P(\text{exactly one girl}) = \frac{n}{2^n}.$$

Now $E$ and $F$ are independent if and only if

$$P(E \cap F) = P(E)P(F),$$

or

$$\frac{n}{2^n} = \left(1 - \frac{1}{2^{n-1}}\right)\left(\frac{n+1}{2^n}\right).$$

This equation simplifies to $2^{n-1} = n + 1$, whose only solution is $n = 3$. (By inspection, $2^{n-1} \neq n + 1$ if $n = 1, 2$. For $n > 3$, $2^{n-1} > n + 1$.) Therefore $E$ and $F$ are independent if and only if $n = 3$.

32. $\dfrac{C(10,5)}{2^{10}}$ 33. $\dfrac{C(10,4) + C(10,5) + C(10,6)}{2^{10}}$

35. $\dfrac{C(10,0) + C(10,1) + C(10,2) + C(10,3) + C(10,4) + C(10,5)}{2^{10}}$

36. Let $E$ be the event "exactly five heads," and let $F$ be the event "at least one head." We want to compute

$$P(E \,|\, F) = \frac{P(E \cap F)}{P(F)}.$$

Now

$$P(E \cap F) = P(E) = \frac{C(10,5)}{2^{10}},$$

and

$$P(F) = 1 - P(\text{no heads}) = 1 - \frac{1}{2^{10}}.$$

Therefore

$$P(E \,|\, F) = \frac{P(E \cap F)}{P(F)} = \frac{\frac{C(10,5)}{2^{10}}}{1 - \frac{1}{2^{10}}}.$$

38. Let $E$ be the event "at least one head," and let $F$ be the event "at least one tail." We want to compute

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)}.$$

Now

$$P(E \cap F) = 1 - P(\text{all heads or all tails}) = 1 - \frac{2}{2^{10}},$$

and

$$P(F) = 1 - P(\text{no tails}) = 1 - \frac{1}{2^{10}}.$$

Therefore

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)} = \frac{1 - \frac{2}{2^{10}}}{1 - \frac{1}{2^{10}}}.$$

39. Let $E$ be the event "at most five heads," and let $F$ be the event "at least one head." We want to compute

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)}.$$

Now

$$\begin{aligned} P(E \cap F) &= P(1 \text{ or } 2 \text{ or } 3 \text{ or } 4 \text{ or } 5 \text{ heads}) \\ &= \frac{C(10,1) + C(10,2) + C(10,3) + C(10,4) + C(10,5)}{2^{10}}, \end{aligned}$$

and

$$P(F) = 1 - P(\text{no heads}) = 1 - \frac{1}{2^{10}}.$$

Therefore

$$P(E \mid F) = \frac{P(E \cap F)}{P(F)} = \frac{\frac{C(10,1)+C(10,2)+C(10,3)+C(10,4)+C(10,5)}{2^{10}}}{1 - \frac{1}{2^{10}}}.$$

41. 253

42. Let $E_1$ denote the event "at least three people have birthdays on the same month and date," let $E_2$ denote the event "exactly two people have birthdays on the same month and date," and let $E_3$ denote the event "no two people have birthdays on the same month and date." Then

$$P(E_1) = 1 - P(\overline{E_1}) = 1 - P(E_2 \text{ or } E_3) = 1 - P(E_2) - P(E_3).$$

The event $E_2$ can be constructed by choosing two people to have a common birthday [in $C(n,2)$ ways]; selecting the date on which they have the same birthday (365 ways); choosing non-common birthdays for the remaining $n-1$ people [$364 \cdot 363 \cdots (364 - n + 3)$ ways]. Thus

$$P(E_2) = \frac{C(n,2) \cdot 365 \cdot 364 \cdots (364 - n + 3)}{365^n}.$$

From Example 6.6.7,

$$P(E_3) = \frac{365 \cdot 364 \cdots (365 - n + 1)}{365^n}.$$

It follows that

$$P(E_1) = 1 - \frac{C(n,2) \cdot 365 \cdot 364 \cdots (364 - n + 3)}{365^n} - \frac{365 \cdot 364 \cdots (365 - n + 1)}{365^n}.$$

43. Let $E_1$ denote the event "at least three people have birthdays on the same month and date." Using the formula in the solution to Exercise 42, we find that for $n = 35$,

$$P(E_1) = 0.480722,$$

and for $n = 36$,

$$P(E_1) = 0.511802.$$

Thus the answer is $n = 36$.

45. Let $H$ be the event "has headache," and let $F$ be the event "has fever." We are given

$$P(H) = 0.01, \quad P(F \mid H) = 0.4, \quad P(F) = 0.02.$$

Using Bayes' Theorem, we have

$$P(H \mid F) = \frac{P(F \mid H)P(H)}{P(F)} = \frac{(0.4)(0.01)}{0.02} = 0.2.$$

47. $P(B \mid A) = 0.01$, $P(B \mid D) = 0.03$, $P(B \mid N) = 0.03$

48.

$$\begin{aligned} P(A \mid B) &= \frac{P(B \mid A)P(A)}{P(B \mid A)P(A) + P(B \mid D)P(D) + P(B \mid N)P(N)} \\ &= \frac{(0.01)(0.55)}{(0.01)(0.55) + (0.03)(0.1) + (0.03)(0.35)} = 0.289473684. \end{aligned}$$

Similarly,

$$P(D \mid B) = 0.157894736, \quad P(N \mid B) = 0.552631578.$$

51. By Theorem 6.6.9,

$$P(E_1 \cap E_2) = P(E_1) + P(E_2) - P(E_1 \cup E_2).$$

Since

$$P(E_1 \cup E_2) \leq 1,$$

the result follows.

52. The Basis Step is $P(E_1) \leq P(E_1)$, which is clearly true.

Assume the statement is true for $n$. By Theorem 6.6.9,

$$\begin{aligned} P(E_1 \cup E_2 \cup \cdots \cup E_n \cup E_{n+1}) &= P(E_1 \cup E_2 \cup \cdots \cup E_n) + P(E_{n+1}) \\ &\quad - P((E_1 \cup E_2 \cup \cdots \cup E_n) \cap E_{n+1}). \end{aligned}$$

Since

$$P((E_1 \cup E_2 \cup \cdots \cup E_n) \cap E_{n+1}) \geq 0,$$

we have

$$P(E_1 \cup E_2 \cup \cdots \cup E_n \cup E_{n+1}) \leq P(E_1 \cup E_2 \cup \cdots \cup E_n) + P(E_{n+1}).$$

Using the inductive assumption, we have

$$\begin{aligned} P(E_1 \cup E_2 \cup \cdots \cup E_n \cup E_{n+1}) &\le P(E_1 \cup E_2 \cup \cdots \cup E_n) + P(E_{n+1}) \\ &\le \sum_{i=1}^{n} P(E_i) + P(E_{n+1}) = \sum_{i=1}^{n+1} P(E_i). \end{aligned}$$

54. Yes. Since $E$ and $F$ are independent, $P(E \cap F) = P(E)P(F)$. Since $E \cap F$ and $E \cap \overline{F}$ are mutually exclusive and $E = (E \cap F) \cup (E \cap \overline{F})$,

$$P(E) = P(E \cap F) + P(E \cap \overline{F}).$$

Now

$$P(E \cap \overline{F}) = P(E) - P(E \cap F) = P(E) - P(E)P(F) = P(E)[1 - P(F)[= P(E)P(\overline{F}).$$

Therefore $E$ and $\overline{F}$ are independent.

55. No. If the person carries a bomb on the plane the probability of a bomb on the plane is 1. The probability of two bombs on the plane is then $1 \cdot 0.000001 = 0.000001$.

## Section 6.7

2. $32c^5 - 240c^4d + 720c^3d^2 - 1080c^2d^3 + 810cd^4 - 243d^5$

4. $59136s^6t^6$ 5. $C(10,2)C(8,3) = 10!/(2!3!5!)$ 7. $C(5,2)$ 8. $C(5,2)$

11. $C(12+4-1,12)$ 12. $C(12+3-1,12) + C(11+3-1,11) + C(10+3-1,10)$

14. (a) $C(n,k) < C(n,k+1)$ if and only if

$$\frac{n!}{k!(n-k)!} < \frac{n!}{(k+1)!(n-k-1)!}$$

if and only if $k+1 < n-k$ if and only if $k < (n-1)/2$.

15. Set $a = 1$ and $b = -1$ in the Binomial Theorem.

17.
$$\begin{aligned} C(n,k-1) + C(n,k) &= \frac{n!}{(k-1)!(n-k+1)!} + \frac{n!}{k!(n-k)!} \\ &= \frac{(n!)k}{k!(n-k+1)!} + \frac{(n!)(n-k+1)}{k!(n-k+1)!} \\ &= \frac{(n!)(n+1)}{k!(n-k+1)!} = C(n+1,k). \end{aligned}$$

18. Choosing a $k$-element set $X$ also selects an $(n-k)$-element set $X$.

20. $(n+1)n(n-1)/3$

21. Use the fact that $k^2 = 2C(k,2) + C(k,1)$.

23. Use Exercise 15 and equation (6.7.3).

24. Imitate the combinatorial proof of the Binomial Theorem.

26. Take $a = b = c = 1$ in Exercise 24.

27. Think of $C(n,k)^2$ as $C(n,k)C(n,n-k)$. Let $X$ and $Y$ be disjoint sets each having $n$ elements. Now, $C(2n,n)$ is the number of ways of picking $n$-element subsets of $X \cup Y$. Picking an $n$-element subset of $X \cup Y$ is the same as picking a $k$-element subset of $X$ and an $(n-k)$-element subset of $Y$.

29. Set $x = 1$ in Exercise 28.

30. Inductive Step.

$$\begin{aligned}
\sum_{k=1}^{n+1} kC(n+1,k) &= \sum_{k=1}^{n} k[C(n,k-1) + C(n,k)] + (n+1)C(n+1,n+1) \\
&= \sum_{k=1}^{n+1} kC(n,k-1) + \sum_{k=1}^{n} kC(n,k) \\
&= \sum_{k=1}^{n+1} (k-1)C(n,k-1) + \sum_{k=1}^{n+1} C(n,k-1) + \sum_{k=1}^{n} kC(n,k) \\
&= n2^{n-1} + 2^n + n2^{n-1} = (n+1)2^n
\end{aligned}$$

32. We count the number of ways to choose sets $A$ and $B$ with $A \subseteq B \subseteq X$. Fix an integer $k$ with $0 \le k \le n$. There are $C(n,k)$ ways to choose a subset $A$ of $X$ with $k$ elements. After choosing such a set $A$, there are $n-k$ elements not in $A$, and there are $2^{n-k}$ ways to choose a subset of them to union with $A$ to produce a set $B$ that contains $A$. Thus there are $C(n,k)2^{n-k}$ ways to choose subsets $A$ and $B$ satisfying $A \subseteq B \subseteq X$ in which $A$ has $k$ elements. Summing over all $k$ we obtain the number of ordered pairs $(A,B)$ satisfying $A \subseteq B \subseteq X$:

$$\sum_{k=0}^{n} C(n,k)2^{n-k}.$$

Taking $a = 2$ and $b = 1$ in the Binomial Theorem, we find that this sum is equal to

$$(a+b)^n = (2+1)^n = 3^n.$$

33. We use induction. The Basis Step is $n = m$:

$$C(m,m)H_m = H_m = C(m+1,m+1)\left(H_{m+1} - \frac{1}{m+1}\right).$$

Assume true for $n$. Then

$$\sum_{k=m}^{n+1} C(k,m)H_k = \sum_{k=m}^{n} C(k,m)H_k + C(n+1,m)H_{n+1}$$

$$\begin{aligned}
&= C(n+1,m+1)\left(H_{n+1}-\frac{1}{m+1}\right)+C(n+1,m)H_{n+1}\\
&= C(n+1,m+1)\left(H_{n+2}-\frac{1}{n+2}-\frac{1}{m+1}\right)\\
&\quad +C(n+1,m)\left(H_{n+2}-\frac{1}{n+2}\right)\\
&= [C(n+1,m+1)+C(n+1,m)]H_{n+2}\\
&\quad -\frac{C(n+1,m+1)+C(n+1,m)}{n+2}-\frac{C(n+1,m+1)}{m+1}\\
&= C(n+2,m+1)H_{n+2}-\frac{C(n+2,m+1)}{n+2}-\frac{C(n+1,m+1)}{m+1}\\
&= C(n+2,m+1)H_{n+2}-\frac{C(n+1,m)}{m+1}-\frac{C(n+1,m+1)}{m+1} \qquad (6.1)\\
&= C(n+2,m+1)H_{n+2}-\frac{C(n+2,m+1)}{m+1}
\end{aligned}$$

Equality (6.1) follows from the formula

$$C(n+2,m+1)=\frac{n+2}{m+1}C(n+1,m).$$

35. Let $x=m/(m+n)$ and $y=n/(m+n)$. The binomial theorem asserts that

$$\left(\frac{m}{m+n}+\frac{n}{m+n}\right)^{m+n}=\sum_{k=0}^{m+n}C(m+n,k)\left(\frac{m}{m+n}\right)^{k}\left(\frac{n}{m+n}\right)^{m+n-k}. \qquad (6.2)$$

Now the left-hand side of equation (6.2) is 1. The term in the right-hand side of equation (6.2) for $k=m$ is

$$C(m+n,m)\left(\frac{m}{m+n}\right)^{m}\left(\frac{n}{m+n}\right)^{n}.$$

Since the other terms on the right-hand side of equation (6.2) are positive, the inequality follows.

36. We call

$$C_k n^k + C_{k-1}n^{k-1}+\cdots+C_1 n+C_0, \quad C_k\neq 0,$$

a *polynomial in $n$ of degree $k$*. The exercise is, then, to show that $\sum_{i=1}^{n} i^k$ is a polynomial in $n$ of degree $k+1$, where the coefficient of $n^{k+1}$ is $1/(k+1)$. We use strong induction on $k$.

Basis Step ($k=0$).

$$\sum_{i=1}^{n} i^k=\sum_{i=1}^{n}1=n$$

Inductive Step. Assume that for $p<k$, $\sum_{i=1}^{n} i^p$ is a polynomial in $n$ of degree $p+1$, where the coefficient of $n^{p+1}$ is $1/(p+1)$.

With $\Delta$ defined as in Exercise 67, Section 2.4, by the binomial theorem

$$\begin{aligned}
\Delta i^{k+1}=(i+1)^{k+1}-i^{k+1} &= [i^{k+1}+(k+1)i^k+C'_{k-1}i^{k-1}+\cdots+C'_1 i+C'_0]-i^{k+1}\\
&= (k+1)i^k+C'_{k-1}i^{k-1}+\cdots+C'_1 i+C'_0.
\end{aligned}$$

By Exercise 67, Section 2.4,

$$\sum_{i=1}^{n} \Delta i^{k+1} = (n+1)^{k+1} - 1 = (k+1)\sum_{i=1}^{n} i^k + C'_{k-1}\sum_{i=1}^{n} i^{k-1} + \cdots + C'_1\sum_{i=1}^{n} i + C'_0\sum_{i=1}^{n} 1.$$

By the inductive assumption,

$$(k+1)\sum_{i=1}^{n} i^k + C'_{k-1}\sum_{i=1}^{n} i^{k-1} + \cdots + C'_1\sum_{i=1}^{n} i + C'_0\sum_{i=1}^{n} 1 = (k+1)\sum_{i=1}^{n} i^k + \text{polynomial in } n \text{ of degree } k.$$

Therefore

$$(n+1)^{k+1} - 1 = (k+1)\sum_{i=1}^{n} i^k + \text{polynomial in } n \text{ of degree } k.$$

Also, by the binomial theorem,

$$\begin{aligned}(n+1)^{k+1} - 1 &= (n^{k+1} + C''_k n^k + \cdots + C''_1 n + 1) - 1 \\ &= n^{k+1} + \text{polynomial in } n \text{ of degree } k.\end{aligned}$$

Therefore

$$n^{k+1} + \text{polynomial in } n \text{ of degree } k = (k+1)\sum_{i=1}^{n} i^k + \text{polynomial in } n \text{ of degree } k.$$

Solving for $\sum_{i=1}^{n} i^k$, we have

$$\begin{aligned}\sum_{i=1}^{n} i^k &= \frac{n^{k+1}}{k+1} + \text{polynomial in } n \text{ of degree } k - \text{polynomial in } n \text{ of degree } k \\ &= \frac{n^{k+1}}{k+1} + \text{polynomial in } n \text{ of degree } \le k.\end{aligned}$$

The Inductive Step is complete.

## Section 6.8

2. Let the six students be the pigeons and the five grades be the pigeonholes. Assign each student (pigeon) to his grade (pigeonhole). By the Pigeonhole Principle, some pigeonhole (grade) will contain at least two pigeons (students), that is, at least two students received the same grade.

3. Let the 32 people be the pigeons and the 31 days in January be the pigeonholes. Assign each person (pigeon) to the day (pigeonhole) he receives a check. By the Pigeonhole Principle, some pigeonhole (day) will contain at least two pigeons (people), that is, at least two people receive checks on the same day.

5. Let the elements of $X$ be the pigeons and the elements of $Y$ be the pigeonholes. Assign each $x \in X$ (pigeon) to $f(x)$ (pigeonhole). By the Pigeonhole Principle, some pigeonhole (value in $Y$) will contain at least two pigeons (elements of $X$), that is, there exist $x_1, x_2 \in X$, $x_1 \neq x_2$, such that $f(x_1) = f(x_2)$; thus $f$ is not one-to-one.

6. Let the six integers selected be the pigeons and the five sets $\{1, 10\}, \{2, 9\}, \{3, 8\}, \{4, 7\}, \{5, 6\}$ be the pigeonholes. Assign each integer selected (pigeon) to the set (pigeonhole) to which it belongs. Notice that since the sets partition $\{1, 2, \ldots, 10\}$, this assignment is unique. By the Pigeonhole Principle, some pigeonhole (set) will contain at least two pigeons (selected integers). Regardless of which set contains two of the selected integers, these integers sum to 11.

8. There are six possible combinations of first and last names. Each of the 18 persons is to be assigned a first and last name. By the Pigeonhole Principle, at least $\lceil 18/6 \rceil = 3$ of them will be assigned the same first and last names.

9. Professor Euclid is paid 26 times per year. Since there are 12 months, by the Pigeonhole Principle, at least $\lceil 26/12 \rceil = 3$ pay periods will occur in the same month.

11. Let $A = \{x_1, \ldots, x_{60}\}$ be the set of positions for the available items. Each $x_i$ assumes a distinct value in $\{1, \ldots, 115\}$. Let $B = \{x_1 + 4, \ldots, x_{60} + 4\}$. The set

$$X = \{x_1, \ldots, x_{60}, x_1 + 4, \ldots, x_{60} + 4\}$$

of 120 numbers can take on values from 1 to 119. By the Pigeonhole Principle at least two of these 120 elements are identical. Since the elements in $A$ are distinct, so are the elements in $B$. There is an element $x_i$ in $A$ and an element in $x_j + 4$ in $B$ which are identical.

12. Let $a_i$ denote the position of the $i$th available item. The 110 numbers

$$a_1, \ldots, a_{55}; \quad a_1 + 9, \ldots, a_{55} + 9$$

have values between 1 and 109. By the second form of the Pigeonhole Principle, two must coincide. The conclusion follows.

14. If any pair $(P_i, P_j)$, $(P_i, P_k)$, $(P_k, P_j)$ is dissimilar, then the two dissimilar pictures together with $P_1$ are three mutually dissimilar pictures. If none of these pairs are dissimilar, then $P_i$, $P_j$, and $P_k$ are three mutually similar pictures.

15. No. Consider five pictures $P_1, \ldots, P_5$ in which $P_1$ is similar to $P_2$; $P_2$ is similar to $P_3$; $P_3$ is similar to $P_4$; $P_4$ is similar to $P_5$; $P_5$ is similar to $P_1$.

16. Yes, since any subset of six pictures has the given property.

21. Let $n = 2$, and consider the subset $\{3, 4, 5\}$ of $\{1, 2, 3, 4, 5\}$.

22. Let $x_i$ denote the longest increasing subsequence and $y_i$ denote the longest decreasing subsequence starting at $a_i$. Consider $(b_i, c_i)$ and $(b_j, c_j)$. We may assume that $i < j$. If $a_i < a_j$, $\{a_i, x_j\}$ is an increasing subsequence starting at $a_i$ which is longer than $x_j$. Hence $b_i \geq$ length of $\{a_i, x_j\} > b_j$. If $a_i > a_j$, $\{a_i, y_j\}$ is a decreasing subsequence starting at $a_i$ which is longer than $y_j$. Hence $c_i \geq$ length of $\{a_i, y_j\} > c_j$. Since the $a_k$'s are distinct, the preceding cases are the only cases. For each, we have shown that $(b_i, c_i)$ and $(b_j, c_j)$ are distinct.

23. The number of ordered pairs $(b_i, c_i)$ is $m = n^2 + 1$, one for each $i = 1, \ldots, m$.

24. By assumption, every increasing or decreasing subsequence has length less than or equal to $n$. Thus $1 \leq b_i \leq n$ and $1 \leq c_i \leq n$.

25. By Exercise 23, we have $n^2 + 1$ pairs $(b_i, c_i)$. By Exercise 24, these pairs can take on only $n^2$ values. By the Pigeonhole Principle, at least two of these pairs must be identical. This contradicts the result of Exercise 22.

26. If $r_i \leq 8$, since $r_i \geq 0$ and $s_i = r_i$, we have $0 \leq s_i \leq 8$.

    If $r_i > 8$, since $r_i \leq 15$, we have $1 \leq 16 - r_i < 8$. Thus $1 \leq s_i < 8$.

27. The set $\{s_1, \ldots, s_{10}\}$ is a subset of the 9-element set $\{0, \ldots, 8\}$. By the second form of the Pigeonhole Principle, $s_j = s_k$ for $j \neq k$.

28. Suppose that $s_j = r_j$ and $s_k = r_k$. Then $r_j = r_k$, so $a_j \bmod 16 = a_k \bmod 16$. Therefore 16 divides $a_j - a_k$.

    If $s_j = 16 - r_j$ and $s_k = 16 - r_k$, we again find that $r_j = r_k$ and the conclusion follows.

29. We may suppose that $s_j = r_j$ and $s_k = 16 - r_k$. Thus $r_j = 16 - r_k$ so $r_j + r_k = 16$. By definition,

$$a_j \bmod 16 = r_j \quad \text{and} \quad a_k \bmod 16 = r_k$$

so there are integers $q_j$ and $q_k$ satisfying

$$a_j = 16q_j + r_j \quad \text{and} \quad a_k = 16q_k + r_k.$$

Now

$$\begin{aligned} a_j + a_k &= 16(q_j + q_k) + r_j + r_k \\ &= 16(q_j + q_k) + 16 \\ &= 16(q_j + q_k + 1). \end{aligned}$$

Therefore 16 divides $a_j + a_k$.

31. Suppose that the numbers around the circle are $x_1, \ldots, x_{12}$. We argue by contradiction. Suppose that

$$\begin{aligned} x_1 + x_2 + x_3 &\leq 19 \\ x_2 + x_3 + x_4 &\leq 19 \\ &\vdots \\ x_{10} + x_{11} + x_{12} &\leq 19 \\ x_{11} + x_{12} + x_1 &\leq 19 \\ x_{12} + x_1 + x_2 &\leq 19. \end{aligned}$$

Summing, we obtain the contradiction

$$234 = 3\left(\frac{12 \cdot 13}{2}\right) = 3(x_1 + \cdots + x_{12}) \leq 12 \cdot 19 = 228.$$

32. The sum of some four consecutive players' numbers must be at least 26. Suppose that

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &\leq 25 \\ x_2 + x_3 + x_4 + x_5 &\leq 25 \\ &\vdots \\ x_{12} + x_1 + x_2 + x_3 &\leq 25 \end{aligned}$$

Summing, we obtain the contradiction

$$312 = 4\left(\frac{12 \cdot 13}{2}\right) = 4(x_1 + \cdots + x_{12}) \leq 12 \cdot 25 = 300.$$

33. Each of the $n! + 1$ functions

$$f, f^2, f^3, \ldots, f^{n!+1}$$

is a permutation of $\{1, \ldots, n\}$. Since there are $n!$ permutations of $\{1, \ldots, n\}$, by the Pigeonhole Principle,

$$f^i = f^j \tag{6.3}$$

for some distinct positive integers $i$ and $j$.

Notice that if we compose each side of (6.3) with $f^{-1}$, we obtain

$$f^{i-1} = f^{j-1}.$$

We may assume that $i > j$. If we compose each side of (6.3) with $f^{-1}$ $j$ times, we obtain

$$f^{i-j} = I,$$

where $I(x) = x$ for $x = 1, \ldots, n$. We may take $k = i - j$ to obtain the desired conclusion.

35. Suppose that the numbers around the circle are $x_1, \ldots, x_{p+q}$. We argue by contradiction. Suppose that each consecutive group of $k$ numbers contains a zero. Then each consecutive group of $k$ numbers sums to $k - 1$ or less. Thus

$$\begin{aligned} x_1 + \cdots + x_k &\leq k - 1 \\ x_2 + \cdots + x_{k+1} &\leq k - 1 \\ &\vdots \\ x_{p+q} + x_1 + \cdots + x_{k-1} &\leq k - 1 \end{aligned}$$

Summing, we obtain

$$kp = k(x_1 + \cdots + x_{p+q}) \leq (p + q)(k - 1)$$

or

$$p \leq (k - 1)q.$$

Since $p \geq kq > (k - 1)q$, this is a contradiction.

36. See Section 6.11.1, pages 167–169, of U. Manber, *Introduction to Algorithms*, Addison-Wesley, Reading, Mass., 1989.

# Chapter 7

# Solutions to Selected Exercises

## Section 7.1

2. $a_n = a_{n-1} + a_{n-2}$; $a_1 = 3$, $a_2 = 6$

3. $a_n = 2a_{n-1}a_{n-2}$; $a_1 = a_2 = 1$

9. $A_n = 1.10A_{n-1} + 2000$

10. $A_0 = 2000$

11. $A_1 = 4200$, $A_2 = 6620$, $A_3 = 9282$

12. $$\begin{aligned} A_n &= 1.10A_{n-1} + 2000 \\ &= 1.10(1.10A_{n-2} + 2000) + 2000 \\ &= 1.10^2 A_{n-2} + (1.10)2000 + 2000 \\ &= 1.10^2(1.10A_{n-3} + 2000) + (1.10)2000 + 2000 \\ &= 1.10^3 A_{n-3} + (1.10^2)2000 + (1.10)2000 + 2000 \\ &\vdots \\ &= 1.10^n A_0 + (1.10^{n-1})2000 + (1.10^{n-2})2000 + \cdots + (1.10)2000 + 2000 \\ &= (1.10^n)2000 + (1.10^{n-1})2000 + (1.10^{n-2})2000 + \cdots + (1.10)2000 + 2000 \\ &= \frac{(1.10^{n+1})2000 - 2000}{1.10 - 1} \\ &= 20000(1.10^{n+1} - 1) \end{aligned}$$

13. $A_n = (1.03)^4 A_{n-1}$

14. $A_0 = 3000$

15. $A_1 = 3376.53$, $A_2 = 3800.31$, $A_3 = 4277.28$

16. $A_n = (1.03)^{4n}3000$

17. 5.86

20. An $n$-bit string that does not contain the pattern 00 either begins 1 and is followed by an $(n-1)$-bit string that does not contain 00, or it begins 01 and is followed by an $(n-2)$-bit string that does not contain 00. Thus we obtain the recurrence relation
$$S_n = S_{n-1} + S_{n-2},$$
which is the same recurrence relation that the Fibonacci sequence $f$ satisfies. The initial conditions for the sequence $S$ are
$$S_1 = 2, \quad S_2 = 3.$$
Since $f_3 = 2$ and $f_4 = 3$, the result follows.

21. We count the number of $n$-bit strings with exactly $i$ 0's that do not contain the pattern 00. Such a string has $n - i$ 1's:

$$\_\_ 1 \_\_ 1 \_\_ \ldots \_\_ 1 \_\_ .$$

To avoid the pattern 00, the 0's must be placed in the $n - i + 1$ spaces. This can be done in $C(n - i + 1, i)$ ways. Thus there are

$$\sum_{i=0}^{\lfloor (n+1)/2 \rfloor} C(n + 1 - i, i) \tag{7.1}$$

$n$-bit strings that do not contain the pattern 00. By Exercise 20, (7.1) is equal to $f_{n+2}$.

23. We count the number of strings not containing 010 having $i$ leading 0's. For $i = 0$, there are $S_{n-1}$ such strings. For $i = 1$, the string begins 011, so there are $S_{n-3}$ such strings. Similarly, for $i = 2$, there are $S_{n-4}$ such strings; ... for $i = n-3$, there are $S_1$ such strings. For $i = n-2$, $n - 1$, or $n$, there is one such string. The equation now follows.

24. The formula for $n - 1$ is

$$S_{n-1} = S_{n-2} + S_{n-4} + S_{n-5} + \cdots + S_1 + 3.$$

Subtracting $S_{n-1}$ from $S_n$, we obtain

$$S_n - S_{n-1} = S_{n-1} + S_{n-3} - S_{n-2}.$$

Solving for $S_n$, we obtain the desired recurrence relation.

26. We use the explicit formula for the $n$th Catalan number derived in Example 6.2.22 to obtain

$$\begin{aligned}
(n+2)C_{n+1} &= \frac{(n+2)C(2n+2, n+1)}{n+2} \\
&= \frac{(2n+2)!}{(n+1)!(n+1)!} \\
&= \frac{2(2n+1)(2n)!(2n+2)}{(n+1)n!n!(2n+2)} \\
&= \frac{(4n+2)(2n)!}{(n+1)n!n!} \\
&= \frac{(4n+2)C(2n,n)}{n+1} = (4n+2)C_n.
\end{aligned}$$

27. We use induction on $n$.

BASIS STEP ($n = 4$). $n + 2 = 6 < 14 = C_4$

INDUCTIVE STEP. Assume that $n+2 < C_n$. Then $(n+2)/C_n < 1$. Using Exercise 26, we have

$$\frac{4n+2}{C_{n+1}} = \frac{n+2}{C_n} < 1.$$

Therefore

$$n + 3 < 4n + 2 < C_{n+1}.$$

29. The proof is by induction on $n$ with the Basis Step omitted.

    Assume that the inequality holds for $n$. We use the formula from Exercise 26 to derive

$$C_{n+1} = \frac{4n+2}{n+2} C_n \geq \frac{4n+2}{n+2} \frac{4^{n-1}}{n^2} \geq \frac{4^n}{(n+1)^2}.$$

    The last inequality is successively equivalent to

$$\begin{aligned} \frac{4n+2}{(n+2)n^2} &\geq \frac{4}{(n+1)^2} \\ (2n+1)(n+1)^2 &\geq 2(n+2)n^2 \\ 2n^3+5n^2+4n+1 &\geq 2n^3+4n^2 \\ n^2+4n+1 &\geq 0, \end{aligned}$$

    which is clearly true for all $n \geq 1$.

30. Let $b_n$ denote the number of ways to parenthesize the product

$$a_1 * \cdots * a_{n+1}.$$

    Then $b_0 = b_1 = 1$.

    Suppose that $n > 1$, $1 \leq i \leq n$, and that the multiplication is carried out by multiplying

$$a_1 * \cdots * a_i, \tag{7.2}$$

    parenthesized in some way, by

$$a_{i+1} * \cdots * a_{n+1}, \tag{7.3}$$

    parenthesized in some way. There are $b_{i-1}$ ways to parenthesize (7.2) and $b_{n-i}$ ways to parenthesize (7.3). Therefore there are $b_{i-1}b_{n-i}$ ways to parenthesize the product

$$a_1 * \cdots * a_{n+1}.$$

    if the multiplication is carried out by multiplying (7.2), parenthesized in some way, by (7.3), parenthesized in some way. Summing over all $i$ we obtain

$$b_n = \sum_{i=1}^{n} b_{i-1}b_{n-i}.$$

    Since the sequence $\{b_n\}$ satisfies the same initial condition and recurrence relation as the Catalan sequence $\{C_n\}$, it follows that $b_n = C_n$ for all $n$.

32. The answer is $(2n)!$. We can order the variables in $(n+1)!$ ways. We can insert the operators in between the variables in $n!$ ways, and, by Exercise 30, we can insert the parentheses in $C_n$ ways. Thus the number of expressions is

$$(n+1)!n!C_n = (n+1)!n!\left[\frac{1}{n+1}C(2n,n)\right] = (n+1)!n!\left[\frac{1}{n+1}\frac{(2n)!}{n!n!}\right] = (2n)!.$$

33. We assume that the paths start at $(0,0)$ and end at $(n,n)$. Suppose that a route first meets the diagonal at $(k+1,k+1)$. It is either always below the diagonal [from $(1,0)$ to $(k+1,k)$] or always above the diagonal [from $(0,1)$ to $(k,k+1)$]. Thus there are $2C_k$ such paths from $(0,0)$ to $(k+1,k+1)$. There are $C(2(n+1-(k+1)),n+1-(k+1)) = C(2(n-k),n-k)$ paths from $(k+1,k+1)$ to $(n+1,n+1)$ (with no restrictions). Thus the number of paths that first meet the diagonal at $(k+1,k+1)$ is $C_kC(2(n-k),n-k)$.

    Since there are $C(2(n+1),n+1)$ paths from $(0,0)$ to $(n+1,n+1)$ (with no restrictions),

$$C(2(n+1),n+1) = \sum_{k=0}^{n} 2C_kC(2(n-k),n-k).$$

    Thus

$$C(2(n+1),n+1) = 2C_n + \sum_{k=0}^{n-1} 2C_kC(2(n-k),n-k).$$

    Dividing by 2 and moving the last summation to the left side of the equation gives the desired result.

35. We assume that the paths start at $(0,0)$ and end at $(n,n)$. Let $D_i$ denote the number of paths that first touch the diagonal at $(i,i)$ after leaving $(0,0)$. Then

$$S_n = D_1 + D_2 + \cdots + D_n.$$

    Since $D_1$ is the product of the number of paths from $(0,0)$ to $(1,1)$ and the number of paths from $(1,1)$ to $(n,n)$,

$$D_1 = 2S_{n-1}.$$

    If $i \geq 2$, $D_i$ is the product of the number of paths from $(1,0)$ to $(i,i-1)$ and the number of paths from $(i,i)$ to $(n,n)$, so

$$D_i = S_{i-1}S_{n-i}.$$

    Thus

$$S_n = 2S_{n-1} + \sum_{i=2}^{n} S_{i-1}S_{n-i}.$$

37. price $= ak/(k+b)$, quantity $= a/(k+b)$

39. We have

$$|p_{n+1} - p_n| = \left| a - \frac{b}{k}p_n - a + \frac{b}{k}p_{n-1} \right| = \left| \frac{b}{k}(p_{n-1} - p_n) \right| = \frac{b}{k}|p_n - p_{n-1}|.$$

    Now $b > k$, so $b/k > 1$. Thus $|p_{n+1} - p_n| > |p_n - p_{n-1}|$.

41. Basis Step. $A(1,0) = A(0,1) = 2$

    Inductive Step. $A(1,n+1) = A(0,A(1,n)) = A(0,n+2) = n+3$

42. Basis Step. $A(2,0) = A(1,1) = 3$ by Exercise 41.

    Inductive Step. Assume that the statement is true for $n$. Then

$$\begin{aligned}A(2,n+1) &= A(1,A(2,n)) \\ &= A(1,3+2n) && \text{by the inductive assumption} \\ &= 3+2n+2 && \text{by Exercise 41} \\ &= 2n+5.\end{aligned}$$

44. Basis Step ($m = 0$). $A(0,n) = n+1 > n$ for all $n \geq 0$.

Inductive Step. Assume that $A(m,n) > n$ for all $n \geq 0$. We use induction on $n$ to prove that

$$A(m+1,n) > n \quad \text{for all } n \geq 0.$$

Basis Step ($n = 0$).

$$\begin{aligned}A(m+1,n) &= A(m+1,0) \\ &= A(m,1) \\ &> 1 && \text{by the original inductive assumption} \\ &> 0 = n.\end{aligned}$$

Inductive Step.

$$\begin{aligned}A(m+1,n+1) &= A(m,A(m+1,n)) \\ &> A(m+1,n) && \text{by the original inductive assumption} \\ &> n && \text{by the present inductive assumption.}\end{aligned}$$

Now

$$A(m+1,n+1) > A(m+1,n) \geq n+1,$$

which completes the current inductive step. Therefore

$$A(m+1,n) > n$$

for all $n \geq 0$. Thus the original induction is complete.

45. If $n = 0$, $A(m,0) = A(m-1,1) > 1$ by Exercise 44.

If $n > 0$, $A(m,n) > n \geq 1$ by Exercise 44.

48. Basis Step ($x = 2$). $AO(2,2,1) = 2 \cdot 1 = 2$.

Inductive Step. Assume that the statement is true for $x$. Now

$$AO(x+1,2,1) = AO(x,2,AO(x+1,2,0)) = AO(x,2,1) = 2.$$

49. Inductive Step.

$$\begin{aligned}AO(x+1,2,2) &= AO(x,2,AO(x+1,2,1)) \\ &= AO(x,2,2) && \text{by Exercise 48} \\ &= 4\end{aligned}$$

51. (a) $a_2 = 1$ because two nodes must establish one link to share files.

Suppose that we have three nodes $A$, $B$, and $C$. If the following successive links are established,

$$A \leftrightarrow B, \quad A \leftrightarrow C, \quad A \leftrightarrow B,$$

then all nodes know all files. Since three links suffice, $a_3 \leq 3$.

Suppose that we have four nodes $A$, $B$, $C$, and $D$. If the following successive links are established,

$$A \leftrightarrow B, \quad C \leftrightarrow D, \quad A \leftrightarrow C, \quad B \leftrightarrow D,$$

then all nodes know all files. Since four links suffice, $a_4 \leq 4$.

(b) Suppose that we have $n \geq 3$ nodes. Let $A$ and $B$ be nodes. First $A$ and $B$ share files. Next, all nodes except $A$ share files (requiring $a_{n-1}$ links). Finally, $A$ and $B$ again share files. At this point all nodes know all files. Thus $a_n \leq a_{n-1} + 2$.

52. $P_1 = 1$, $P_n = nP_{n-1}$

54. Suppose that we have $n$ dollars. If we buy tape the first day, there are $R_{n-1}$ ways to spend the remaining money. If we buy paper the first day, there are $R_{n-1}$ ways to spend the remaining money. If we buy pens the first day, there are $R_{n-2}$ ways to spend the remaining money. If we buy pencils the first day, there are $R_{n-2}$ ways to spend the remaining money. If we buy binders the first day, there are $R_{n-3}$ ways to spend the remaining money. Thus

$$R_n = 2R_{n-1} + 2R_{n-2} + R_{n-3}.$$

55. Because of the assumptions, when the $(n+1)$st line $L$ is added, it will intersect the other $n$ lines. If we imagine traveling along $L$, each time we pass through one of the original regions, we divide it into two regions. Since we pass through $n+1$ regions, $R_{n+1} = R_n + n + 1$.

57. $S_n = \frac{2}{3}\left[1 - \left(-\frac{1}{2}\right)^{n-1}\right]$

59. A string $\alpha$ of length $n$ with $C(\alpha) \leq 2$ either begins with 1 (there are $S_{n-1}$ of these), 01 (there are $S_{n-2}$ of these), or 001 (there are $S_{n-3}$ of these). Thus $S_n = S_{n-1} + S_{n-2} + S_{n-3}$.

60. Basis Steps ($n = 1, 2$).

$$2f_2 - 1 = 2 - 1 = 1 = g_1, \quad 2f_3 - 1 = 4 - 1 = 3 = g_2$$

Inductive Step.

$$g_n = g_{n-1} + g_{n-2} + 1 = (2f_n - 1) + (2f_{n-1} - 1) + 1 = 2(f_n + f_{n-1}) - 1 = 2f_{n+1} - 1$$

62. The problem is that the Inductive Step assumes *two* previous cases, but the Basis Step proves only *one*.

63. $C(n+1, k) = C(n, k-1) + C(n, k)$

65. There are $k^n$ functions from $X = \{1, \ldots, n\}$ onto $Y = \{1, \ldots, k\}$. We will count the number $N$ of functions from $X$ into, but not onto, $Y$. Then, the number of functions from $X$ onto $Y$ will be $k^n - N$.

    Let $Z$ be a proper, nonempty subset of $Y$ with $i$ elements. There are $S(n, i)$ functions from $X$ onto $Z$. The number of subsets of $Y$ having $i$ elements is $C(k, i)$. Then, there are $C(k, i)S(n, i)$ functions from $X$ onto some $i$-element subset of $Y$. If a function from $X$ to $Y$ is not onto $Y$, it is onto some proper nonempty subset of $Y$. The result follows.

66. (a) $L_3 = 4$, $L_4 = 7$, $L_5 = 11$

    (b) Basis Step ($n = 1, 2$).

$$\begin{aligned} L_3 &= 4 = 1 + 3 = f_2 + f_4 \\ L_4 &= 7 = 2 + 5 = f_3 + f_5 \end{aligned}$$

    Inductive Step. Assume that $L_{n+1} = f_n + f_{n+2}$ and $L_{n+2} = f_{n+1} + f_{n+3}$. Now

$$\begin{aligned} L_{n+3} &= L_{n+2} + L_{n+1} = f_{n+1} + f_{n+3} + f_n + f_{n+2} \\ &= (f_n + f_{n+1}) + (f_{n+2} + f_{n+3}) \\ &= f_{n+2} + f_{n+4}. \end{aligned}$$

68. Let $X$ be an $(n + 1)$-element set and choose an element $x \in X$. We count the number of partitions of $X$ containing $k$ subsets in which $x$ appears as a singleton and the number of partitions of $X$ containing $k$ subsets in which $x$ appears in a subset with at least two elements.

    A partition of $X$ containing $k$ subsets in which $x$ appears as a singleton consists of $\{x\}$ together with a partition of $X - \{x\}$ containing $k - 1$ subsets. There are $S_{n,k-1}$ such partitions.

    A partition of $X$ containing $k$ subsets in which $x$ appears in a subset with at least two elements can be constructed in the following way. Select a partition of $X - \{x\}$ containing $k$ subsets. This can be done in $S_{n,k}$ ways. Next add $x$ to one of the subsets. This can be done in $k$ ways. Thus there are $kS_{n,k}$ partitions of $X$ containing $k$ subsets in which $x$ appears in a subset with at least two elements. The recurrence relation now follows.

69. We prove the formula by induction on $n$ leaving the Basis Step ($n = 1$) to the reader.

    Assume that the formula is true for $n$. We must prove the formula is true for $n + 1$. If $k = 0$, the formula is clearly true, so assume that $k > 0$. By Exercise 68,

$$\begin{aligned} S_{n+1,k} &= S_{n,k-1} + kS_{n,k} \\ &= \frac{1}{(k-1)!} \sum_{i=0}^{k-1} (-1)^i (k-1-i)^n C(k-1, i) + \frac{k}{k!} \sum_{i=0}^{k} (-1)^i (k-i)^n C(k, i) \\ &= \frac{1}{(k-1)!} \left[ \sum_{i=0}^{k-1} (-1)^i (k-1-i)^n C(k-1, i) + \sum_{i=0}^{k} (-1)^i (k-i)^n C(k, i) \right] \\ &= \frac{1}{(k-1)!} \left[ k^n C(k, 0) + \sum_{i=1}^{k} \{(-1)^{i-1} (k-i)^n C(k-1, i-1) + (-1)^i (k-i)^n C(k, i)\} \right] \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{(k-1)!}\left[k^n C(k,0) + \sum_{i=1}^{k}(-1)^i (k-i)^n[-C(k-1,i-1)+C(k,i)]\right] \\
&= \frac{1}{(k-1)!}\left[k^n C(k,0) + \sum_{i=1}^{k}(-1)^i (k-i)^n C(k-1,i)\right] \\
&= \frac{1}{(k-1)!}\left[\frac{k^{n+1}C(k,0)}{k} + \sum_{i=1}^{k}(-1)^i (k-i)^n \frac{C(k,i)(k-i)}{k}\right] \\
&= \frac{1}{k!}\sum_{i=0}^{k}(-1)^i (k-i)^{n+1} C(k,i).
\end{aligned}
$$

71. $a_n = n(a_{n-1}+1)$

73. 1,5,2,4,3; 1,5,3,4,2; 2,5,1,4,3; 2,5,3,4,1; 3,5,1,4,2; 3,5,2,4,1; 4,5,1,3,2; 4,5,2,3,1; 1,3,2,5,4; 2,3,1,5,4; 1,4,2,5,3; 2,4,1,5,3; 1,4,3,5,2; 3,4,1,5,2; 2,4,3,5,1; 3,4,2,5,1; $E_5 = 16$

74. An item in the first, third, ... position has a larger neighbor; therefore, $n$ cannot be in any of these positions.

76. The solution is similar to that of Exercise 75, which is given in the book, except that the portion following 1 must be a "fall/rise" permutation. However, the number of "fall/rise" permutations of $1,\ldots,n$ is equal to the number of "rise/fall" permutations of $1,\ldots,n$, so the argument proceeds as in the solution to Exercise 75.

77. Add the recurrence relations of Exercises 75 and 76.

## Section 7.2

2. No 3. No 5. Yes; order 3 6. No 8. Yes; order 2

9. Yes; order 2 12. $a_n = 2^n n!$

13.
$$
\begin{aligned}
a_n &= a_{n-1} + n = a_{n-2} + (n-1) + n = \cdots \\
&= a_0 + 1 + 2 + \cdots + n = 1 + 2 + \cdots + n = \frac{n(n+1)}{2}
\end{aligned}
$$

14. $a_n = 2^n a_{n-1} = 2^n 2^{n-1} a_{n-2} = 2^n 2^{n-1} 2^{n-2} a_{n-3} = \cdots = 2^n 2^{n-1} 2^{n-2} \cdots 2^1 a_0$
$= 2^{n+(n-1)+\cdots+1} = 2^{n(n+1)/2}$

16. Solving $t^2 - 7t + 10 = (t-2)(t-5)$, we obtain two roots $t = 2$ and $t = 5$. Thus there exist constants $b$ and $d$ such that $a_n = b2^n + d5^n$. The initial conditions require that $5 = b + d$ and $16 = 2b + 5d$. Solving these two equations simultaneously for $b$ and $d$, we obtain $b = 3$ and $d = 2$. Thus

$$a_n = 3 \cdot 2^n + 2 \cdot 5^n.$$

17. The roots of

$$t^2 - 2t - 8 = 0$$

are 4 and $-2$. Thus the solution is $a_n = b4^n + d(-2)^n$. To satisfy the initial conditions, we must have

$$\begin{aligned} 4 &= b + d \\ 10 &= 4b - 2d. \end{aligned}$$

Solving, we find $b = 3$ and $d = 1$. Thus the solution is

$$a_n = 3 \cdot 4^n + (-2)^n.$$

19. $$\begin{aligned} a_n &= a_{n-1} + 1 + 2^{n-1} \\ &= (a_{n-2} + 1 + 2^{n-2}) + 1 + 2^{n-1} \\ &= a_{n-2} + 2 + 2^{n-1} + 2^{n-2} = \cdots \\ &= a_0 + n + 2^{n-1} + 2^{n-2} + \cdots + 1 = n + 2^n - 1 \end{aligned}$$

20. $a_n = 3^n - 2n3^{n-1}$

22. $a_n = 6\left(\frac{1}{3}\right)^n + 9n\left(\frac{1}{3}\right)^n$

23. Similar to Example 7.2.13

25. $s_n = \dfrac{n(n+1)}{2} + 1$

26. $S_n = \frac{2}{3}\left[1 - \left(-\frac{1}{2}\right)^{n-1}\right]$

27. Let $p$ be the population of Utopia in 1970. Arguing as in Example 7.1.3, we find that $n$ years after 1970, the population of Utopia is $(1.05)^n p$. Therefore $10000 = (1.05)^{30} p$. Solving for $p$, we find that the population of Utopia in 1970 was $p = 2314$.

30. 0

31. 1

33. $\dfrac{T}{S+T}$

34. The recurrence relation becomes $b_n = b_{n-1} + 2b_{n-2}$. Solving gives $a_n = b_n^2 = \frac{1}{9}\left[2^{n+1} + (-1)^n\right]^2$.

35. Taking the logarithm to the base 2 of both sides of the equation, we obtain

$$\lg a_n = \frac{1}{2}(\lg a_{n-2} - \lg a_{n-1}).$$

If we let $b_n = \lg a_n$, we obtain

$$b_n = \frac{-b_{n-1}}{2} + \frac{b_{n-2}}{2}.$$

The quadratic equation

$$t^2 + \frac{t}{2} - \frac{1}{2} = 0$$

has two roots, $\frac{1}{2}$ and $-1$. Thus there exist constants $p$ and $q$ such that

$$b_n = p\left(\frac{1}{2}\right)^n + q(-1)^n.$$

Now

$$b_0 = \lg a_0 = \lg 8 = 3$$

and a similar calculation shows that $b_1 = -\frac{3}{2}$. Therefore

$$\begin{aligned} 3 &= p + q \\ -\frac{3}{2} &= \frac{p}{2} - q. \end{aligned}$$

Solving for $p$ and $q$, we obtain $p = 1$ and $q = 2$. Thus

$$b_n = \left(\frac{1}{2}\right)^n + 2(-1)^n$$

and

$$a_n = 2^{b_n} = 2^{[(1/2)^n + 2(-1)^n]}.$$

37. Subtracting the given recurrence relation from the recurrence relation for $n + 1$

$$c_{n+1} = 2 + \sum_{i=1}^{n} c_i.$$

gives

$$c_{n+1} - c_n = c_n$$

so

$$c_{n+1} = 2c_n, \quad n \geq 2.$$

This last recurrence relation can be solved by iteration to yield

$$c_{n+1} = 2c_n = 2^2 c_{n-1} = \cdots = 2^{n-1} c_2 = 3 \cdot 2^{n-1}.$$

for $n \geq 2$. By inspection, the formula also holds for $n = 1$. Thus we obtain the formula

$$c_n = 3 \cdot 2^{n-2},$$

for $n \geq 2$.

38. Let $S(n,m) = A(n,m) - C(n,m) + 1$. [$C(n,m)$ is the number of $m$-element subsets of an $n$-element set.] Then $S(n,n) = 1 = S(n,0)$. Also

$$\begin{aligned} S(n-1,m-1) + S(n-1,m) &= A(n-1,m-1) - C(n-1,m-1) + 1 \\ &\quad + A(n-1,m) - C(n-1,m) + 1 \\ &= A(n,m) - C(n,m) + 1 = S(n,m). \end{aligned}$$

Since $\{S(n,m)\}$ and $\{C(n,m)\}$ satisfy the same recurrence relation and have the same initial conditions, they are equal. Therefore

$$A(n,m) = S(n,m) + C(n,m) - 1 = 2C(n,m) - 1.$$

40. Show that $U_n - g(n)$ satisfies

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}.$$

42. Assume that

$$g(n) = C_1 n + C_0$$

is a solution. We must have

$$C_1 n + C_0 = 7[C_1(n-1) + C_0] - 10[C_1(n-2) + C_0] + 16n.$$

The coefficient of $n$ on the left must equal the coefficient of $n$ on the right:

$$C_1 = 7C_1 - 10C_1 + 16.$$

Thus $C_1 = 4$.

The constant on the left must equal the constant on the right:

$$C_0 = -7C_1 + 7C_0 + 20C_1 - 10C_0.$$

Thus $C_0 = 13$. Therefore

$$g(n) = 4n + 13.$$

The general solution of

$$a_n = 7a_{n-1} - 10a_{n-2}$$

is

$$b2^n + d5^n.$$

Thus the general solution of the original recurrence relation is

$$a_n = b2^n + d5^n + 4n + 13.$$

43. Assume that

$$g(n) = C_2 n^2 + C_1 n + C_0$$

is a solution. We must have

$$C_2 n^2 + C_1 n + C_0 = 2[C_2(n-1)^2 + C_1(n-1) + C_0] + 8[C_2(n-2)^2 + C_1(n-2) + C_0] + 81n^2.$$

The coefficient of $n^2$ on the left must equal the coefficient of $n^2$ on the right:

$$C_2 = 2C_2 + 8C_2 + 81.$$

Thus $C_2 = -9$.

The coefficient of $n$ on the left must equal the coefficient of $n$ on the right:

$$C_1 = -4C_2 + 2C_1 - 32C_2 + 8C_1.$$

Thus $C_1 = -36$.

The constant on the left must equal the constant on the right:

$$C_0 = 2C_2 - 2C_1 + 2C_0 + 32C_2 - 16C_1 + 8C_0.$$

Thus $C_0 = -38$. Therefore

$$g(n) = -9n^2 - 36n - 38.$$

The general solution of

$$a_n = 2a_{n-1} + 8a_{n-2}$$

is

$$a_n = b4^n + d(-2)^n.$$

Thus the general solution of the original recurrence relation is

$$a_n = b4^n + d(-2)^n - 9n^2 - 36n - 38.$$

45. $a_n = b4^n + dn4^n + \frac{3}{25}n + \frac{24}{125}$

46. $a_n = b(1/3)^n + dn(1/3)^n + (5/4)n^2 - (5/2)n + 25/8$

48. We must have

$$\begin{aligned} C_0 &= a_0 = b \\ C_1 &= a_1 = b + d. \end{aligned}$$

Set $b = C_0$ and $d = C_1 - C_0$.

49. By Exercise 51, Section 7.1, we have

$$a_n \le a_{n-1} + 2, \quad n \ge 4; \quad a_4 \le 4.$$

Now

$$\begin{aligned} a_n &\le a_{n-1} + 2 \le a_{n-2} + 2 + 2 < \cdots \\ &\le a_{n-i} + 2i \le \cdots \\ &\le a_4 + 2(n-4) \le 4 + 2(n-4) = 2n - 4. \end{aligned}$$

51.

| $n$ | $T(n)$ | *Opt Moves for 3-Peg Problem* |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 3 | 3 |
| 3 | 5 | 7 |
| 4 | 9 | 15 |
| 5 | 13 | 31 |
| 6 | 17 | 63 |
| 7 | 25 | 127 |
| 8 | 33 | 255 |
| 9 | 41 | 511 |
| 10 | 49 | 1023 |

52. We show only the inductive step. Using the recurrence relation of Exercise 50 and the inductive assumption, we have

$$T(n) = 2T(n-k_n) + 2^{k_n} - 1 = 2[(k_{n-k_n} + r_{n-k_n} - 1)2^{k_{n-k_n}} + 1] + 2^{k_n} - 1.$$

First suppose that

$$n - k_n < \sum_{i=1}^{k_n} i.$$

Since

$$\sum_{i=1}^{k_n} i \le n,$$

it follows that

$$\sum_{i=1}^{k_n - 1} i \le n - k_n.$$

Therefore,

$$k_{n-k_n} = k_n - 1.$$

Also,

$$r_n = n - \sum_{i=1}^{k_n} i = n - k_n - \sum_{i=1}^{k_n - 1} i = r_{n-k_n}.$$

Now

$$\begin{aligned} T(n) &= 2[(k_{n-k_n} + r_{n-k_n} - 1)2^{k_{n-k_n}} + 1] + 2^{k_n} - 1 \\ &= 2[(k_n - 1 + r_n - 1)2^{k_n - 1} + 1] + 2^{k_n} - 1 \\ &= (k_n - 1 + r_n - 1)2^{k_n} + 2 + 2^{k_n} - 1 \\ &= (k_n + r_n - 1)2^{k_n} + 1. \end{aligned}$$

The case

$$n - k_n = \sum_{i=1}^{k_n} i$$

is treated similarly. (In this case, $k_{n-k_n} = k_n$ and $r_{n-k_n} = 0$.)

54. We call a stacking in which the disks are arranged from top to bottom in order from smallest to largest a *proper stacking*. We call a stacking in which the disks in arbitrary order except that the largest disk is on the bottom an *arbitrary stacking*.

Now consider the given problem at the point when the bottom disk first moves. There must be an empty peg for it to move to; thus, the remaining $n - 1$ disks must be optimally moved to a third peg in an arbitrary stacking. We first determine the minimum number of moves required to move disks from a proper stacking to another peg in an arbitrary stacking.

Our new problem is, given $n$ disks in a proper stacking, find the minimum number of moves, which we denote $s_n$, to move these $n$ disks to another peg in an arbitrary stacking. Except for the original position, arbitrary stackings are allowed.

Clearly, $s_1 = 1$. Suppose $n > 1$. Consider the point at which the bottom disk is moved. One peg must be empty (to receive the largest disk), and the $n-1$ smaller disks must be moved to a third peg. By definition, this move requires $s_{n-1}$ moves. Now the largest disk moves (which requires one additional move). Now the $n-1$ smaller disks must be placed on top of the largest disk. We can simply peel them off one-by-one and place them on top of the largest disk. This requires $n-1$ moves, which is surely optimal. Therefore,

$$s_n = s_{n-1} + 1 + n - 1.$$

Solving, we obtain

$$s_n = 1 + 2 + \cdots + n = \frac{n(n+1)}{2}.$$

Now we can answer the original question. Consider the point at which the bottom disk first moves. There must be an empty peg for it to move to; thus, the remaining disks must be moved to a third peg, which requires $n(n+1)/2$ moves. Now the largest disk moves (which requires one additional move). Finally, the smallest $n-1$ disks must be optimally moved and properly stacked on top of the largest disk. This can be done by *reversing* the first $n(n+1)/2$ moves. [If there were some way to move the smallest $n-1$ disks and properly stack them on top of the largest disk in fewer than $n(n+1)/2$, we could reverse this technique and obtain a method of moving from a proper stacking to an arbitrary stacking in fewer than $n(n+1)/2$ moves.] Thus optimum number of moves is

$$\frac{n(n+1)}{2} + 1 + \frac{n(n+1)}{2} = n(n+1) + 1.$$

## Section 7.3

2. At line 2, since $i > j$ ($1 > 5$) is false, we proceed to line 4 where we set $k$ to 3. At line 5, since *key* ($'P'$) is not equal to $s_3$ ($'J'$), we proceed to line 7. At line 7, $key < s_k$ ($'P' < 'J'$) is false, so at line 10, we set $i$ to 4. We then invoke this algorithm with $i = 4, j = 5$ to search for *key* in

$$s_4 = 'M',\ s_5 = 'X'.$$

At line 2, since $i > j$ ($4 > 5$) is false, we proceed to line 4 where we set $k$ to 4. At line 5, since *key* ($'P'$) is not equal to $s_4$ ($'M'$), we proceed to line 7. At line 7, $key < s_k$ ($'P' < 'M'$) is false, so at line 10, we set $i$ to 5. We then invoke this algorithm with $i = j = 5$ to search for *key* in

$$s_5 = 'X'.$$

At line 2, since $i > j$ ($5 > 5$) is false, we proceed to line 4 where we set $k$ to 5. At line 5, since *key* ($'P'$) is not equal to $s_5$ ($'X'$), we proceed to line 7. At line 7, $key < x_k$ ($'P' < 'X'$) is true, so at line 8, we set $j$ to 4. We then invoke this algorithm with $i = 5, j = 4$ to search for *key* in an empty list.

At line 2, since $i > j$ ($5 > 4$) is true, we return 0 to signal an unsuccessful search.

3. At line 2, since $i > j$ $(1 > 5)$ is false, we proceed to line 4 where we set $k$ to 3. At line 5, since *key* $('C')$ is not equal to $s_3$ $('J')$, we proceed to line 7. At line 7, $key < s_k$ $('C' < 'J')$ is true, so at line 8, we set $j$ to 2. We then invoke this algorithm with $i = 1, j = 2$ to search for *key* in

$$s_1 = 'C',\ s_2 = 'G'.$$

At line 2, since $i > j$ $(1 > 2)$ is false, we proceed to line 4 where we set $k$ to 1. At line 5, since *key* $('C')$ is equal to $s_1$ $('C')$, we return 1, the index of *key* in the sequence $s$.

5. We give a proof using induction. We omit the Basis Step.

Assume that $a_i \le a_{i+1}$ for all $i < n$. We must prove the inequality for $n$.

Using (7.3.2) and the inductive assumption, we have

$$a_n = 1 + a_{\lfloor n/2 \rfloor} \le 1 + a_{\lfloor (n+1)/2 \rfloor} = a_{n+1}.$$

6. We use induction on $n$. The Basis Step $(n = 1)$ is omitted.

INDUCTIVE STEP. Suppose that $n > 1$ and $a_k = \lfloor \lg k \rfloor + 2$ for all $k < n$.

If $n$ is odd,

$$\begin{array}{rcll} a_n & = & 1 + a_{\lfloor n/2 \rfloor} = 1 + a_{(n-1)/2} & \\ & = & 1 + \lfloor \lg(n-1)/2 \rfloor + 2 & \text{by the inductive assumption} \\ & = & 3 + \lfloor \lg(n-1) - 1 \rfloor & \\ & = & 3 + \lfloor \lg(n-1) \rfloor - 1 & \lfloor x - 1 \rfloor = \lfloor x \rfloor - 1 \\ & = & 2 + \lfloor \lg(n-1) \rfloor & \\ & = & 2 + \lfloor \lg n \rfloor & \text{if } j \text{ is odd, } \lfloor \lg j \rfloor = \lfloor \lg(j-1) \rfloor. \end{array}$$

The case when $n$ is even is treated similarly.

8. No. The only way that the algorithm finds *key* is when the condition

$key == s_k$

evaluates to true.

9. This algorithm searches for *key* in the nondecreasing sequence $s_i, \ldots, s_j$, $i \ge 1$. If *key* is found, the algorithm returns an index $k$ such that $s_k$ equals *key*. If *key* is not found, the algorithm returns 0.

```
bsearch_nonrecurs(s, i, j, key) {
   while (i ≤ j) {
      k = ⌊(i + j)/2⌋
      if (key == s_k) // found
         return k
      if (key < s_k) // search left half
         j = k − 1
      else // search right half
```

```
        i = k + 1
    }
    return 0 // not found
}
```

11. The idea is to run the algorithm of Exercise 10 twice. Suppose that the values returned are $i$ and $j$. Since only one incorrect response is allowed, either $i$ or $j$ is the correct value. If $i = j$, then $i$ (or $j$) is the correct value. If $i < j$, we can compare $s_j$ with *key*. If $s_j = key$, then $j$ is the correct value; otherwise, $i$ is the correct value. Similarly, If $j < i$, we can compare $s_i$ with *key*. If $s_i = key$, then $i$ is the correct value; otherwise, $j$ is the correct value.

    Running the algorithm of Exercise 10 twice uses at most $2(1+\lceil \lg n \rceil) = 2+2\lceil \lg n \rceil$ comparisons. Since one additional comparison may be required, our algorithm uses and most $3 + 2\lceil \lg n \rceil$ comparisons.

12. (a) We use induction on the size $n$ of the sequence to prove that *binary_search2* is correct when the input is a sequence of size $n$.

    The Basis Step is $n = 0$. In this case, the sequence is empty, $i > j$, and the algorithm correctly returns 0 to indicate that *key* is not found.

    Now assume that if a sequence of length less than $n$ is input to *binary_search2*, the algorithm returns the correct value. Suppose that a sequence of length $n > 0$ is input to *binary_search2*. Since $i \le j$, the algorithm proceeds to the line where it computes $k$. If *key* is at index $k$, the algorithm correctly returns $k$. If *key* is not at index $k$, since the sequence is sorted, *key*, if present, is either in $s_i, \ldots, s_{k-1}$ or $s_{k+1}, \ldots, s_j$, but not both. The algorithm then executes

    $k1 = binary\_search2(s, i, k-1, key)$

    If *key* is present in $s_i, \ldots, s_{k-1}$, by the inductive assumption, *binary_search2* returns the index where it is located. If *key* is not present in $s_i, \ldots, s_{k-1}$, by the inductive assumption, *binary_search2* returns 0. The value returned is stored in $k1$. The algorithm then executes

    $k2 = binary\_search2(s, k+1, j, key)$

    If *key* is present in $s_{k+1}, \ldots, s_j$, by the inductive assumption, *binary_search2* returns the index where it is located. If *key* is not present in $s_{k+1}, \ldots, s_j$, by the inductive assumption, *binary_search2* returns 0. The value returned is stored in $k2$. It follows that if *key* is present in $s_i, \ldots, s_{k-1}$ or $s_{k+1}, \ldots, s_j$, it is at index $k1 + k2$. If *key* is not present in $s_i, \ldots, s_j$, $k1 + k2 = 0$. Since the algorithm returns $k1 + k2$, it follows that the algorithm is correct.

    (b) We define the worst-case time required by the algorithm to be the number of times the algorithm is invoked in the worst case for a sequence containing $n$ items. Let let $a_n$ denote the worst-cast time.

    Suppose that $n$ is 0, that is, $i > j$. In this case, there is one invocation; so $a_0 = 1$.

    Now suppose that $n > 1$. In the worst case, the item will not be found at the line

    if ($key == s_k$)

    so the algorithm will be invoked twice more:

    $k1 = binary\_search2(s, i, k-1, key)$
    $k2 = binary\_search2(s, k+1, j, key)$

By definition, the first invocation will require a total of $a_{\lfloor (n-1)/2 \rfloor}$ invocations, and the second invocation will require a total of $a_{\lfloor n/2 \rfloor}$ invocations. Thus we obtain the recurrence relation

$$a_n = 1 + a_{\lfloor (n-1)/2 \rfloor} + a_{\lfloor n/2 \rfloor}.$$

We use strong induction to show that

$$a_n \leq 3n + 1$$

for all $n \geq 0$, thus proving that $a_n = O(n)$. The Base Case, $n = 0$, is readily verified. Now assume that $n > 0$. By the inductive assumption,

$$a_{\lfloor (n-1)/2 \rfloor} \leq 3\lfloor (n-1)/2 \rfloor + 1 \qquad \text{and} \qquad a_{\lfloor n/2 \rfloor} \leq 3\lfloor n/2 \rfloor + 1.$$

Now

$$\begin{aligned} a_n &= 1 + a_{\lfloor (n-1)/2 \rfloor} + a_{\lfloor n/2 \rfloor} \\ &\leq 1 + 3\lfloor (n-1)/2 \rfloor + 1 + 3\lfloor n/2 \rfloor + 1 \\ &= 3 + 3(\lfloor (n-1)/2 \rfloor + \lfloor n/2 \rfloor) \\ &= 3 + 3(n-1) = 3n < 3n + 1. \end{aligned}$$

We conclude by using strong induction to show that

$$a_n \geq n$$

for all $n \geq 0$, thus proving that $a_n = \Omega(n)$ and, therefore, $a_n = \Theta(n)$. The Base Case, $n = 0$, is readily verified. Now assume that $n > 0$. By the inductive assumption,

$$a_{\lfloor (n-1)/2 \rfloor} \geq \lfloor (n-1)/2 \rfloor \qquad \text{and} \qquad a_{\lfloor n/2 \rfloor} \geq \lfloor n/2 \rfloor.$$

Now

$$\begin{aligned} a_n &= 1 + a_{\lfloor (n-1)/2 \rfloor} + a_{\lfloor n/2 \rfloor} \\ &\geq 1 + \lfloor (n-1)/2 \rfloor + \lfloor n/2 \rfloor \\ &= 1 + (n-1) = n. \end{aligned}$$

14. The algorithm is correct. The worst-case time is $\Theta(n)$.

15. The algorithm is not correct. The input $s_1 = 10$ and $key = 8$ results in infinite recursion.

17. The algorithm is not correct. If $i = j$, the index $k$ is out of bounds.

19.

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 9 | 9 | 3 |
| 7 | 3 | 7 |
| 3 | 7 | 9 |
| Merge one-element arrays | Merge two-element arrays | |

22. We give a recursive description. We assume that the input consists of the integers from 1 to $n$. Let $m = \lfloor (n+1)/2 \rfloor$.

    Set
$$s_1 = 1, s_2 = 3, s_3 = 5, \ldots, s_m = 2m - 1$$
    and
$$s_{m+1} = 2, s_{m+2} = 4, \ldots, s_n = 2\lfloor n/2 \rfloor.$$
    Now arrange $s_1, \ldots, s_m$ to produce worst-case behavior for *merge_sort* and arrange $s_{m+1}, \ldots, s_n$ to produce worst-case behavior for *merge_sort*.

23. Seven, which occurs for an already sorted array.

25. Basis Step. $a_1 = 0 < 2 = 2a_1 + 2 = a_2$.

    Inductive Step. Assume that the inequality holds for $k < n$. Now
$$\begin{aligned} a_{n+1} &= a_{\lfloor (n+1)/2 \rfloor} + a_{\lfloor (n+2)/2 \rfloor} + n \\ &\geq a_{\lfloor n/2 \rfloor} + a_{\lfloor (n+1)/2 \rfloor} + n - 1 = a_n. \end{aligned}$$

26. The last inequality in the proof of Theorem 7.3.10 gives $a_n \leq 2n \lg n + 2n + 1$ for all $n$. If $3 \leq \lg n$ (or, equivalently, if $8 \leq n$), $2n + 1 \leq 3n \leq n \lg n$. Therefore if $8 \leq n$,
$$a_n \leq 2n \lg n + 2n + 1 \leq 3n \lg n.$$
    The cases $1 \leq n < 7$ can be checked directly.

27. Sequences of length $m$ and $n$, where $m \leq n$, require, at the minimum, $m - 1$ comparisons for merging. Thus if $b_n$ denotes the least number of comparisons used by merge sort, $b_n$ satisfies
$$b_n = b_{\lfloor n/2 \rfloor} + b_{\lfloor (n+1)/2 \rfloor} + \left\lfloor \frac{n}{2} \right\rfloor - 1.$$
    This recurrence relation can be estimated in the same way as the one for the worst-case time; therefore, the best-case time is $\Theta(n \lg n)$.

33. If $n = 1$, $a^n = a$; thus, $a$ is returned. For $n > 1$, if $n$ is even, $m = \lfloor n/2 \rfloor = n/2$ and $a^n = a^m a^m$. If $n$ is odd, $m = (n-1)/2$ and $a^n = a^m a^m a$.

34. If $n$ is odd, $m = (n-1)/2$. If $n > 1$, after $a^m$ is computed in line 5, line 6 is executed to compute $a^m a^m$ and line 10 is then executed to compute $a^m a^m a$. Thus $b_n = b_{(n-1)/2} + 2$. If $n$ is even, $m = n/2$ and line 6 is executed to compute $a^m a^m$. In this case, line 10 is not executed, so $b_n = b_{n/2} + 1$.

35. $b_1 = 0,\ b_2 = 1,\ b_3 = 2,\ b_4 = 2$

36. Assume that $n = 2^k$. Now

$$b_n = b_{2^{k-1}} + 1 = b_{2^{k-2}} + 2 = \cdots = b_1 + k = 0 + k = \lg n.$$

37. $b_7 = 4$, $b_8 = 3$

38. We use induction on $n$ to prove that

$$\lg n \le b_n \le 2 \lg n,$$

from which we deduce $b_n = \Theta(\lg n)$. The Basis Step ($n = 1$) is omitted.

Assume that $n > 1$. If $n$ is even, we have

$$b_n = b_{n/2} + 1 \le 1 + 2 \lg \frac{n}{2} = 1 + 2[(\lg n) - 1] \le 2 \lg n$$

and

$$b_n = b_{n/2} + 1 \ge 1 + \lg \frac{n}{2} = 1 + (\lg n) - 1 = \lg n.$$

If $n$ is odd, we have

$$b_n = b_{(n-1)/2} + 2 \le 2 + 2 \lg \frac{n-1}{2} = 2 + [2 \lg(n-1)] - 2 \le 2 \lg n$$

and

$$b_n = b_{(n-1)/2} + 2 \ge 2 + \lg \frac{n-1}{2} = 2 + [\lg(n-1)] - 1 = 1 + \lg(n-1) \ge \lg n.$$

The last inequality is equivalent to

$$1 \ge \lg \frac{n}{n-1}$$

or

$$2 \ge \frac{n}{n-1}$$

which is easily seen to be true for $n > 1$.

39. If $i = j$, there is only one element in the array, which is both largest and smallest. In this case, the algorithm simply returns these values.

If $i < j$, the algorithm divides the array into two nearly equal parts at line 7. At lines 8 and 9, the algorithm recursively finds that largest and smallest elements in each of the parts. The overall largest element is the larger of the largest in each of the parts (computed at line 11 or 13), and the overall smallest element is the smaller of the smallest in each of the parts (computed at line 15 or 17).

40. For input of size 1, $i = j$; no comparisons are made since the algorithm returns at line 5. Thus $b_1 = 0$.

    For input of size 2, no comparisons are made during the recursive calls at lines 8 and 9 (since each involves input of size 1). There is one comparison at line 10 and one at line 14. Thus $b_2 = 2$.

41. 4

42. At lines 8 and 9, $a_{\lfloor n/2 \rfloor} + a_{\lfloor (n+1)/2 \rfloor}$ comparisons are made. At lines 10 and 14, two comparisons are made.

43. If $n = 2^k$, the recurrence relation becomes $b_{2^k} = 2b_{2^{k-1}} + 2$. Now

$$\begin{aligned} b_n &= b_{2^k} = 2b_{2^{k-1}} + 2 = 2[2b_{2^{k-2}} + 2] + 2 \\ &= 2^2 b_{2^{k-2}} + 2^2 + 2 = \cdots \\ &= 2^k b_{2^0} + 2^k + 2^{k-1} + \cdots + 2 \\ &= 2^k + 2^{k-1} + \cdots + 2 = 2^{k+1} - 2 = 2 \cdot 2^k - 2 = 2n - 2 \end{aligned}$$

44. We give only the Inductive Step.

    Assume that $b_k = 2k - 2$, for $k < n$. The inductive assumption gives

$$\begin{aligned} b_{\lfloor n/2 \rfloor} &= 2\lfloor n/2 \rfloor - 2 \\ b_{\lfloor (n+1)/2 \rfloor} &= 2\lfloor (n+1)/2 \rfloor - 2. \end{aligned}$$

    If $n$ is even, $\lfloor n/2 \rfloor = n/2 = \lfloor (n+1)/2 \rfloor$, so

$$b_n = 2[2(n/2) - 2] + 2 = 2n - 2.$$

    If $n$ is odd, $\lfloor n/2 \rfloor = (n-1)/2$ and $\lfloor (n+1)/2 \rfloor = (n+1)/2$, so

$$b_n = [2(n-1)/2 - 2] + [2(n+1)/2 - 2] + 2 = 2n - 2.$$

50. If $n = 1$, there is nothing to sort so the algorithm simply returns. If $n > 1$, the elements $s_1, \ldots, s_{n-1}$ are sorted as a result of the recursive call. To sort the entire array, the $n$th element $s_n$, stored in the variable *temp*, is compared to each of the preceding elements which are moved up one position successively as long as they are greater than *temp*. The index $i$ runs down the list. As soon as an element less than or equal to *temp* is found (pointed to by $i$), the loop is exited and *temp* is stored at position $i + 1$. If there is no element less than or equal to *temp*, $i$ is 0 and *temp* is stored at position 1.

51. The worst-case behavior occurs when the items are in reverse order.

52. $b_1 = 0,\ b_2 = 1,\ b_3 = 3$

53. $b_n = b_{n-1} + n - 1$

54. $b_n = \dfrac{n(n-1)}{2}$

55. $b_n = 1 + \lfloor \lg n \rfloor + b_{\lfloor n/2 \rfloor}, b_1 = 1, b_2 = 3, b_3 = 3$

56. $b_n = (1 + \lg n)(2 + \lg n)/2$

57. An arbitrary value of $n$ falls between two powers of 2, say

$$2^{k-1} < n \le 2^k.$$

This inequality implies that $k - 1 < \lg n \le k$. Since the sequence $b$ is nondecreasing,

$$b_{2^{k-1}} \le b_n \le b_{2^k}.$$

Now

$$b_n \le b_{2^k} = \frac{(1+k)(2+k)}{2} \le \frac{(2+\lg n)(3+\lg n)}{2} = O((\lg n)^2).$$

Similarly, $b_n = \Omega((\lg n)^2)$. Therefore $b_n = \Theta((\lg n)^2)$.

60. $b_1 = 1$, $b_2 = 2$, $b_3 = 3$,

61. $b_n = b_{n-1} + 1$

62. $b_n = n$

63. $\Theta(n)$

65. Similar to the proof for Exercise 57

67. $b_n = b_{\lfloor (1+n)/2 \rfloor} + b_{\lfloor n/2 \rfloor} + n$

68. Let $c_k = b_{2^k}$. Then $c_k = 2c_{k-1} + 3$. If $n = 2^k$,

$$\begin{aligned} b_n &= c_k = 2c_{k-1} + 3 = 2(2c_{k-2} + 3) + 3 = \cdots \\ &= 2^k c_0 + 3(2^{k-1} + 2^{k-2} + \cdots + 1) = 2^k \cdot 0 + 3(2^k - 1) \\ &= 3(n-1). \end{aligned}$$

70. Use the method of Exercise 68.

71. $b_n = n(1 + \lg n)$

73. Use Exercise 70 to show that if $n$ is a power of 2, $b_n = n \lg n$. Now let $n$ be arbitrary. Choose $k$ so that $2^k < n \le 2^{k+1}$. By Exercise 72, $b_n \le b_{2^k+1}$. Now

$$b_{2^{k+1}} = 2^{k+1}(k+1) \le 2^{k+1}(k+k) = 4(2^k k) \le 4n \lg n.$$

75. $a_n \le a_{\lfloor n/2 \rfloor} + a_{\lfloor (n+1)/2 \rfloor} + 2 + \lg(\lfloor n/2 \rfloor \lfloor (n+1)/2 \rfloor)$. If $n$ is even, $\lfloor n/2 \rfloor \lfloor (n+1)/2 \rfloor = n^2/4$. If $n$ is odd, $\lfloor n/2 \rfloor \lfloor (n+1)/2 \rfloor = [(n-1)/2][(n+1)/2] = (n^2-1)/4 \le n^2/4$. Thus we can write

$$a_n \le a_{\lfloor n/2 \rfloor} + a_{\lfloor (n+1)/2 \rfloor} + 2 + \lg \frac{n^2}{4} = a_{\lfloor n/2 \rfloor} + a_{\lfloor (n+1)/2 \rfloor} + 2 \lg n.$$

76. Let $n = 2^k$ and use induction on $k$.

BASIS STEP. If $k = 0$, $b_1 = 0 = 4 \cdot 1 - 2 \lg 1 - 4$

INDUCTIVE STEP.

$$\begin{aligned} b_{2^{k+1}} &= 2b_{2^k} + 2(k+1) = 2[4 \cdot 2^k - 2k - 4] + 2(k+1) \\ &= 4 \cdot 2^{k+1} - 2(k+1) - 4 \end{aligned}$$

78. INDUCTIVE STEP.

$$b_n = b_{\lfloor n/2 \rfloor} + b_{\lfloor (n+1)/2 \rfloor} + 2 \lg n \le b_{\lfloor (n+1)/2 \rfloor} + b_{\lfloor (n+2)/2 \rfloor} + 2 \lg(n+1) = b_{n+1}$$

79. Choose $k$ with $2^k < n \le 2^{k+1}$. Then

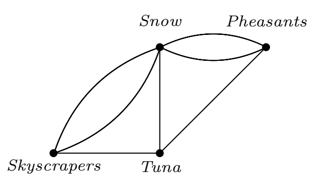
$$a_n \le b_n \le b_{2^{k+1}} = 4 \cdot 2^{k+1} - 2(k+1) - 4 \le 8 \cdot 2^k \le 8n.$$

# Chapter 8

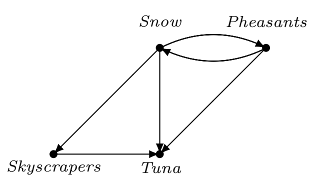
# Solutions to Selected Exercises

## Section 8.1

2. The following undirected graph models the tournament:




3. The following simple, directed graph models the tournament:




6. There are vertices of odd degree $\{b, d\}$.

7. There are vertices of odd degree $\{b, d\}$.

9. $(a, c, f, e, c, b, e, d, b, a)$

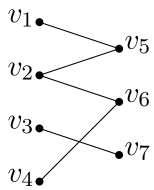
10. $(a, b, c, e, b, d, e, f, c, g, h, i, f, h, e, g, d, a)$

12. $V = \{v_1, v_2, v_3, v_4, v_5\}$. $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$. There are no parallel edges, no loops, and no isolated vertices. $G$ is simple. $e_1$ is incident on $v_2$ and $v_4$.

13. $V = \{v_1, v_2, v_3\}$. $E$ is empty. There are no parallel edges and no loops. $G$ is simple. $e_1$ does not exist. All vertices are isolated.

15. $n(n-1)/2$

16. $V_1 = \{v_1, v_2, v_3, v_4\}$, $V_2 = \{v_5, v_6, v_7\}$.




18. The graph is bipartite. $V_1 = \{v_1, v_3, v_4, v_6, v_8, v_9, v_{10}\}$, $V_2 = \{v_2, v_5, v_7\}$.

19. The graph is bipartite. $V_1 = \{Gre, Buf, Sho, Dou, Mud\}$, $V_2 = \{She, Wor, Cas, Gil, Lan\}$.

21. Not bipartite 22. Not bipartite 25. $mn$

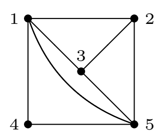
26. Example 8.1.12 is bipartite, but Examples 8.1.13 and 8.1.14 are not bipartite. $K_1$ is not bipartite because there is no way to partition the vertices into two *nonempty* subsets.

28. $(c, a, b, e, d)$ 29. $(a, c, d, e, b)$

30. The vertices are mathematicians, and an edge connects two mathematicians if they co-authored a paper. The Erdős number of mathematician $m$ is the length of a shortest path from $m$ to Erdős.

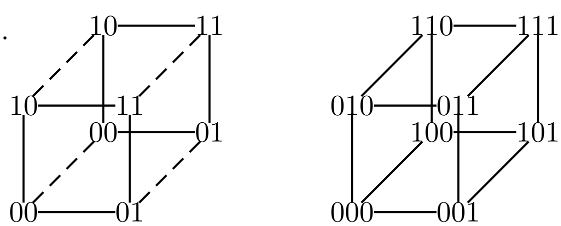
31. Yes

33. One class.




34. No. If similarity is defined solely by the dissimilarity function, in the graph of Figure 8.1.8, $v_1$ is similar to $v_3$ and $v_3$ is similar to $v_5$, but $v_1$ is not similar to $v_5$.

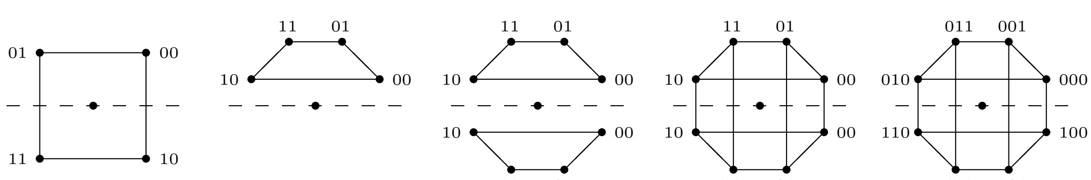
38.




41. $n2^{n-1}$ (There are $2^n$ vertices and each is incident on $n$ edges. Since $n2^n$ counts each edge twice, the formula follows.)

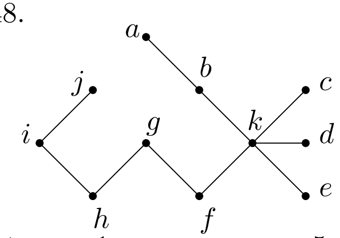
42. $n!2^n$ (The result can be proved by induction on $n$.)

44.

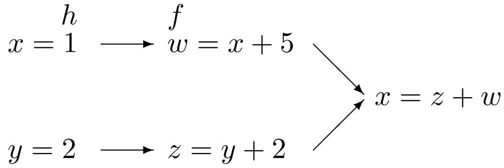


47. 5

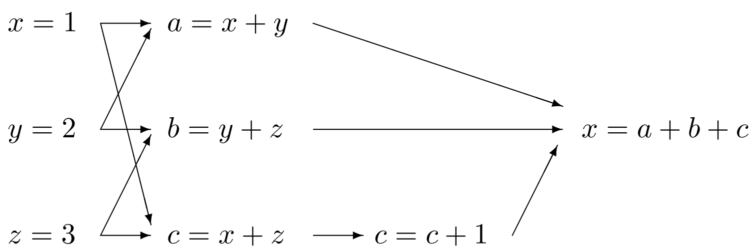
48.




51.




52.




## Section 8.2

2. Simple path–yes; cycle–no; simple cycle–no

3. Simple path–no; cycle–no; simple cycle–no

5. Simple path–no; cycle–no; simple cycle–no

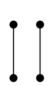
6. Simple path–no; cycle–yes; simple cycle–no

8. Simple path–yes; cycle–no; simple cycle–no

9. Simple path–yes; cycle–no; simple cycle–no

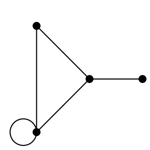
11. There is no such graph since there are always an even number of vertices of odd degree.

12.

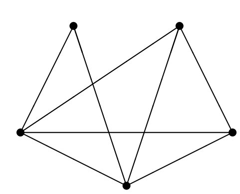


14. No such graph exists. One-half the sum of the degrees (= number of edges) is 5, not 4.

15.



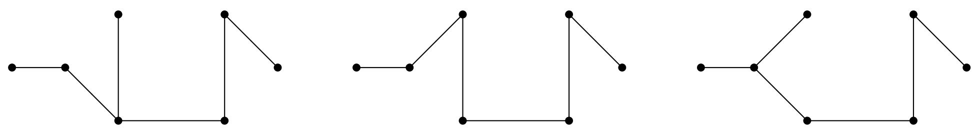
17.



18. No such graph exists. Suppose, by way of contradiction, that there is such a graph with vertices $a$ and $b$ of degree 2 and $c$, $d$, and $e$ of degree 4. Since $c$ is of degree 4, it is incident on $a$, $b$, $d$, and $e$. Similarly, $d$ is incident on $a$, $b$, $c$, and $e$, and $e$ is incident on $a$, $b$, $c$, and $d$. But now $a$ has degree at least 3. Contradiction.

20. $(a, b, c, d, e)$, $(a, b, c, d, f, e)$, $(a, b, c, g, f, e)$, $(a, b, c, g, f, d, e)$, $(a, b, g, f, e)$, $(a, b, g, c, d, e)$, $(a, b, g, f, d, e)$, $(a, b, g, c, d, f, e)$

21.

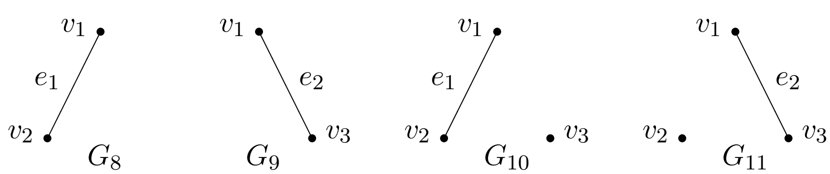


The second is a simple path. Neither is a cycle or a simple cycle.

23. $\delta(v_1) = 2, \delta(v_2) = 2, \delta(v_3) = 3, \delta(v_4) = 6, \delta(v_5) = 2, \delta(v_6) = 3, \delta(v_7) = 4, \delta(v_8) = 4, \delta(v_9) = 4, \delta(v_{10}) = 2$

25. There are six subgraphs, the following five subgraphs and the original graph itself.


26. The following are the subgraphs with no edges: $G_1 = (\{v_1\}, \emptyset)$, $G_2 = (\{v_2\}, \emptyset)$, $G_3 = (\{v_3\}, \emptyset)$, $G_4 = (\{v_1, v_2\}, \emptyset)$, $G_5 = (\{v_1, v_3\}, \emptyset), G_6 = (\{v_2, v_3\}, \emptyset), G_7 = (\{v_1, v_2, v_3\}, \emptyset)$. The other subgraphs are




and $G$ itself.

29. $(v_1, v_5, v_2, v_4, v_5, v_3, v_2, v_1, v_4, v_3, v_1)$

30. There is no Euler cycle since there are vertices of odd degree.

32. $(a, b, d, c, b, f, g, j, f, e, j, h, c, i, d, e, i, h, a)$

33. $(a, b, c, b, d, e, h, f, i, j, k, i, h, g, f, e, g, d, c, a)$

35. When $n$ is odd

36. Both $m$ and $n$ must be even.

38. When $n$ is even

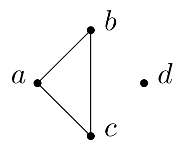
40. There are zero vertices of even degree, and zero is even.

41. $(d, a, b, d, e, b, c, e, h, g, d, f, g, j, h, i, e)$

43. $(b, c, d, g, b, a, f, g, i, f)$; $(c, h, e)$

44. Let $G$ be a connected graph with $n$ vertices $v_1, \ldots, v_n$ of odd degree. There are paths with no repeated edges from $v_1$ to $v_2$, $v_3$ to $v_4$, and so on, such that every edge in $G$ is in exactly one of the paths.

46. False. Consider the cycle $(a, b, c, a)$ for the graph




48. The graph of Exercise 26

49. No. Let the vertices of a graph be the squares of the chessboard. Insert an edge between two vertices if a knight can make a move between the corresponding squares. The degrees of the vertices that correspond to the border squares next to the corner squares have degree 3. Since there are eight of these, there is no Euler cycle.

51. Let $H$ be one of the connected subgraphs in the partition. Let $v$ be a vertex in $H$. Let $C$ be the component to which $v$ belongs. We show that $H = C$.

    Let $w$ be a vertex in $H$. Since $H$ is connected, there is a path from $v$ to $w$ in $H$. Therefore $w$ is in $C$.

    Let $w$ be a vertex in $C$. Then there is a path from $v$ to $w$

$$(v_0, v_1, \ldots, v_n),$$

    with $v_0 = v,\ v_n = w$, in $C$. The edge $(v_0, v_1)$ must belong to $H$ since vertex $v_0$ is in $H$. Thus vertex $v_1$ is in $H$. Continuing in this way, we see that $w$ is in $H$. Therefore the vertex sets of $H$ and $C$ are equal.

    Similarly, the edges sets of $H$ and $C$ are equal. Therefore the subgraphs of the partition are components.

52. There is a path $P$ from $v$ to $w$. Change the orientation of each edge in $P$.

54. Let $G$ be a simple, bipartite graph having the maximum number of edges with disjoint vertex sets having $k$ and $n-k$ vertices. Then $G$ has $k(n-k)$ edges. The maximum of the integer-valued function $f(k) = k(n-k)$ occurs when $k = n/2$, if $n$ is even, and when $k = (n-1)/2$ or $k = (n+1)/2$, if $n$ is odd. Thus the maximum number of edges is $\lfloor n^2/4 \rfloor = n^2/4$, if $n$ is even, and $\lfloor n^2/4 \rfloor = [(n-1)/2][(n+1)/2]$, if $n$ is odd.

56. $K_6$

60. If

$$(b_1^{(1)} b_2^{(1)} \ldots b_{n-1}^{(1)}, b_1^{(2)} b_2^{(2)} \ldots b_{n-1}^{(2)}, \ldots)$$

    is a directed Euler cycle in G,

$$b_1^{(1)} b_2^{(1)} \ldots b_{n-1}^{(1)} b_{n-1}^{(2)} b_{n-1}^{(3)} \ldots$$

    is a de Bruijn sequence.

62. We first show that if $G$ is a connected bipartite graph, then every closed path in $G$ has even length.

Suppose that the disjoint vertex sets are $V_1$ and $V_2$. Let

$$P = (v_0, v_1, \ldots, v_n)$$

be a closed path from $v_0$ to $v_n$. Suppose that $v_0 \in V_1$. Then $v_1 \in V_2,\ v_2 \in V_1, \ldots$. Notice that if $i$ is odd, $v_i \in V_2$, and if $i$ is even, $v_i \in V_1$. Since $v_n \in V_1$, it follows that $n$ is even. Thus $P$ has even length.

We conclude by showing that if every closed path in a connected graph $G$ has even length, then $G$ is bipartite.

Choose a vertex $v$ in $G$. Let $V_1$ denote the set of vertices $w$ in $G$ that are reachable from $v$ on a path of even length. Let $V_2$ denote the set of vertices $w$ in $G$ that are reachable from $v$ on a path of odd length. Notice that since $G$ is connected, every vertex in $G$ is either in $V_1$ or $V_2$. We claim that $V_1$ and $V_2$ are disjoint. To show this, we argue by contradiction. Suppose that some vertex $w$ belongs to both $V_1$ and $V_2$. Then there is a path $P_o'$ of odd length and a path $P_e$ of even length from $v$ to $w$. Let $P_o$ be the path from $w$ to $v$ obtained by reversing the order of the vertices. Then $P_e$ followed by $P_o$ is a closed path of odd length from $v$ to $v$. This contradiction shows that $V_1$ and $V_2$ are disjoint. Now let $e$ be an edge incident on vertices $x$ and $y$. Suppose that $x$ belongs to $V_1$. Then there is a path $P$ of even length from $v$ to $x$. Now $P$ followed by $y$ is a path of odd length from $v$ to $y$. Thus $y$ is in $V_2$. Therefore $G$ is bipartite.

63. $n(n-1)^k$ [Choose $(v_0, v_1, \ldots, v_k)$ with $v_{i-1} \neq v_i$.]

65. (a) $p_m = (n-1)^{m-1} - p_{m-1}$. The first term counts the number of paths of length $m-1$ that start with $v$, and the last term counts the number of paths of length $m-1$ that start with $v$ and end with $w$. A path of length $m-1$ that starts with $v$ and does not end with $w$ can be extended to a path of length $m$ that starts with $v$ and ends with $w$.

(b)

$$\begin{aligned}
p_m &= (n-1)^{m-1} - [(n-1)^{m-2} - p_{m-2}] \\
&= (n-1)^{m-1} - (n-1)^{m-2} + p_{m-2} \\
&\vdots \\
&= (n-1)^{m-1} - (n-1)^{m-2} + \cdots + (-1)^m(n-1) + (-1)^{m+1}p_1 \\
&= (n-1)^{m-1} - (n-1)^{m-2} + \cdots + (-1)^m(n-1) + (-1)^{m+1} \\
&= \frac{-(n-1)^m - (-1)^{m+1}}{-(n-1) - 1} \\
&= \frac{(n-1)^m + (-1)^{m+1}}{n}
\end{aligned}$$

66. There is one path of length 1, $(v, w)$, from $v$ to $w$.

There are $n-2$ paths of length 2, $(v, x_1, w)$, from $v$ to $w$ since vertex $x_1$ can be chosen in $n-2$ ways. (Vertex $x_1$ must be different from $v$ and $w$.)

There are $(n-2)(n-3)$ paths of length 3, $(v, x_1, x_2, w)$, from $v$ to $w$ since vertex $x_1$ can be chosen in $n-2$ ways, and vertex $x_2$ can be chosen in $n-3$ ways. (Vertex $x_1$ must be different from $v$ and $w$, and vertex $x_2$ must be different from $v$, $w$, and $x_1$.)

In general, there are $(n-2)(n-3)\cdots(n-k)$ paths of length $k$, $(v, x_1, \ldots, x_{k-1}, w)$, from $v$ to $w$ since vertex $x_1$ can be chosen in $n-2$ ways, vertex $x_2$ can be chosen in $n-3$ ways, and so on. The result now follows.

67. The number of simple paths of length 0 is $n$; the number of simple paths of length 1 is $n(n-1)$; the number of simple paths of length 2 is $n(n-1)(n-2)$; and so on. Thus the number of simple paths is

$$n + n(n-1) + n(n-1)(n-2) + \cdots + n(n-1)\cdots 1 = n! \sum_{k=0}^{n-1} \frac{1}{k!}.$$

Now

$$\begin{aligned} n! \sum_{k=0}^{n-1} \frac{1}{k!} &= n! \left( e - \sum_{k=n}^{\infty} \frac{1}{k!} \right) \\ &= n!e - 1 - n! \sum_{k=n+1}^{\infty} \frac{1}{k!}. \end{aligned}$$

Since, for $n \geq 2$,

$$\begin{aligned} n! \sum_{k=n+1}^{\infty} \frac{1}{k!} &= \frac{1}{n+1} + \frac{1}{(n+1)(n+2)} + \cdots \\ &< \frac{1}{n} + \frac{1}{n^2} + \cdots = \frac{\frac{1}{n}}{1 - \frac{1}{n}} = \frac{1}{n-1} \leq 1, \end{aligned}$$

the result follows. The result is true by inspection for $n = 1$.

69. A connected graph with one vertex, consists of the vertex, say $v$, and none or more loops incident on $v$. An Euler cycle consists of a cycle that traverses each loop once.

A connected graph with two vertices, say $v$ and $w$, each of which has even degree, consists of $2k$ edges incident on $v$ and $w$, $k \geq 1$, none or more loops incident on $v$, and none or more loops incident on $w$. An Euler cycle consists of a path that begins at $v$, traverses all of the loops incident on $v$, traverses one edge from $v$ to $w$, traverses all of the loops incident on $w$, traverses one edge from $w$ to $v$, and traverses all remaining edges incident on $v$ and $w$. This path will end at $v$ since there are an even number of edges incident on $v$ and $w$.

71. diameter $= n$. The diameter is the maximum time for two processors to communicate.

72. 1, since $\text{dist}(v, w) = 1$ for every pair of distinct vertices in $K_n$

74. First we show that, if $n \bmod 4 = 1$,

$$k \geq \frac{n-1}{2}.$$

Suppose that $n \bmod 4 = 1$. In particular, $n$ is odd. Since every vertex has degree $k$, $k$ must be even. (If $k$ is odd, we obtain a contradiction to the theorem that states that there are an even number of vertices of odd degree). We show that $(n-3)/2$ is an odd integer and, consequently,

$$k \geq \frac{n-3}{2} + 1 = \frac{n-1}{2}.$$

Since $n \bmod 4 = 1$, we may write $n = 4q + 1$. Thus $n - 3 = 4q - 2$ and $(n-3)/2 = 2q - 1$, which is an odd integer. Therefore

$$k \geq \frac{n-1}{2},$$

regardless of the value of $n \bmod 4$.

Now suppose that $G$ is not connected. Let $C_1$ and $C_2$ be components. Since every vertex has degree $k$, $C_1$ and $C_2$ each have at least $k+1$ vertices. Thus $G$ has at least $2(k+1) \geq n+1$ vertices, which is a contradiction.

76. We prove the result by induction on $n$. We omit the Basis Step ($n = 1$).

    Assume that the result is true for $n$. Let $G$ be an $(n+1)$-vertex dag with the maximum number of edges. By Exercise 75, $G$ has a vertex $v$ with no out edges. In fact, there must be edges of the form $(w, v)$ for all $w \neq v$; otherwise, $G$ would not have the maximum number of edges. This accounts for $n$ edges.

    Let $G'$ be the graph obtained from $G$ by eliminated $v$ and the $n$ edges incident on $v$. $G'$ is an $n$-vertex dag and since $G$ has the maximum number of vertices, $G'$ must also have the maximum number of vertices. By the inductive assumption, $G'$ has $n(n-1)/2$ vertices. Thus $G$ has

$$\frac{n(n-1)}{2} + n = \frac{(n+1)n}{2}$$

    vertices.

77. Let $I(P_n)$ denote the number of independent sets in $P_n$. Note that $I(P_1) = 2$ and $I(P_2) = 3$. Now suppose that $n > 2$. Let $v$ be a vertex of degree 1 in $P_n$. An independent set $P_n$ that contains $v$ consists of $v$ and an independent set of $P_{n-2}$, and there are $I(P_{n-2})$ such independent sets. An independent set of $P_n$ that does not contain $v$ is an independent set of $P_{n-1}$, and there are $I(P_{n-1})$ such independent sets. Therefore

$$I(P_n) = I(P_{n-1}) + I(P_{n-2}).$$

    Since $\{I(P_n)\}$ satisfies the same initial conditions and recurrence relation as $\{f_{n+2}\}$, $I(P_n) = f_{n+2}$ for all $n$.

78. (a) Let $v$ and $w$ be nonadjacent vertices in $G$. Let

$$\{x_1, \ldots, x_k\}$$

    denote the vertices adjacent to $v$. Then the mapping $N(x_i) = y_i$, where $y_i$ is adjacent to $x_i$ and $w$, is a bijection to the set of vertices adjacent to $w$. Therefore $\delta(v) = \delta(w)$.

(b) Let $V_1$ denote the set of vertices of degree $k$. Suppose that $\overline{V_1}$ is nonempty. By part (a), every vertex in $V_1$ is adjacent to every vertex in $\overline{V_1}$. Since no vertex is adjacent to all other vertices, $|V_1| \leq 2$ and $|\overline{V_1}| \leq 2$. Let $v_1$ and $w_1$ be distinct vertices in $V_1$, and let $v_2$ and $w_2$ be distinct vertices in $|\overline{V_1}|$. Now $v_1$ adjacent to $v_2$ and $w_2$ and $w_1$ adjacent to $v_2$ and $w_2$, which is a contradiction since there are two vertices ($v_2$ and $w_2$) adjacent to both $v_1$ and $w_1$.

## Section 8.3

2. $(a, b, c, d, e, f, n, p, m, l, k, j, o, i, h, g, a)$

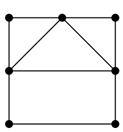
4. We would have to eliminate one edge at $f$, three edges at $c$, one edge at $b$, one edge at $i$, three edges at $j$, and three edges at $m$, leaving 15 edges. Since there are 16 vertices, a Hamiltonian cycle would have 16 edges.

5. Suppose that the graph has a Hamiltonian cycle. Since each vertex in a cycle has degree 2, we would have to include the edges $(a, b)$, $(a, f)$, $(f, g)$, $(b, c)$, $(c, d)$, $(d, e)$, $(e, h)$, and $(g, h)$. Since these edges already form a cycle, there is no Hamiltonian cycle.

7. $(a, b, c, g, l, m, r, q, p, k, j, f, e, i, n, o, t, s, h, d, a)$

8. There is no Hamiltonian cycle. We would have to eliminate two edges at $c$, three edges at $e$, and one edge at $f$, leaving six edges. Since there are seven vertices, a Hamiltonian cycle would have seven edges.

10. $K_3$

11. 

13. We begin the Hamiltonian cycle at row 1, column 1 and proceed along the first row to column $m-1$. Then we go down to row 2 and move back along row 2 to column 1. Then we go down to row 3 and along row 3 to column $m-1$. Then we go down to row 4 and back to column 1 along row 4. We continue this serpentine path until we arrive at the last row. If $n$ is odd, we finish in column 1. We then take the edge from row $n$, column 1 to row $n$, column $m$ and proceed up column $m$ to row 1. We then follow the edge from row 1, column $m$ to row 1, column 1 finishing the cycle. If $n$ is odd, we ended our path at row $n$, column $m - 1$. We then move along row $n$ to column $m$, up column $m$ to row 1, and along the edge from row 1, column $m$ to row 1, column 1 to finish the cycle.

14. Choose any vertex $v$ to start. After arriving at a vertex, move to a not-yet-visited vertex (except when returning to $v$ for the $n$th and last move). Since the degree of every vertex is $n - 1$ and there are $n$ moves, such moves are always possible.

16. The five edges of smallest weight have weights 3, 4, 4, 5, 5. Thus the shortest Hamiltonian cycle has a weight of at least 21. However, three of these edges (those with weights 3, 4, 4) are

incident on vertex $c$. Thus the edges of weight 3, 4, 4 cannot all be in a Hamiltonian cycle. If we replace an edge of weight 4 with an edge of minimum replacement weight 6, we can conclude that the shortest Hamiltonian cycle has weight at least

$$3 + 6 + 4 + 5 + 5 = 23.$$

Since the given Hamiltonian cycle has weight 23, we conclude that it is minimal.

17. $(e, d, a, b, c, e)$

19.

| | |
|---|---|
| $G_1$: | 0 1 |
| $G_1^R$: | 1 0 |
| $G_1'$: | 00 01 |
| $G_1''$: | 11 10 |
| $G_2$: | 00 01 11 10 |
| $G_2^R$: | 10 11 01 00 |
| $G_2'$: | 000 001 011 010 |
| $G_2''$: | 110 111 101 100 |
| $G_3$: | 000 001 011 010 110 111 101 100 |
| $G_3^R$: | 100 101 111 110 010 011 001 000 |
| $G_3'$: | 0000 0001 0011 0010 0110 0111 0101 0100 |
| $G_3''$: | 1100 1101 1111 1110 1010 1011 1001 1000 |
| $G_4$: | 0000 0001 0011 0010 0110 0111 0101 0100 1100 1101 1111 1110 1010 1011 1001 1000 |

20. Let $C$ be a Hamiltonian cycle in $G$. Consider a traversal of $C$. When we traverse an edge from a vertex $v_1$ in $V_1$ to a vertex $v_2$ in $V_2$, this uniquely associates one vertex $v_2$ with $v_1$. Since $C$ traverses all vertices $|V_1| = |V_2|$.

22. Let each vertex of a graph represent a permutation. Put an edge between two vertices $p$ and $q$ if and only if $p_i \neq q_i$ for all $i = 1, 2, \ldots, n$.

23. $(n = 1)$ 1

$(n = 2)$ 12 21

$(n = 3)$ Consider a graph with six vertices representing the permutations and with an edge between two vertices if the permutations differ in each coordinate. A solution to the problem is a Hamiltonian path in the graph. There is no solution for $n = 3$ because the graph is not connected.

$(n = 4)$ 1234 3412 4321 1432 2341 4132 3241 4123 3214 1423 2314 1243 2134 3421 4312 2431 1342 4231 3142 4213 3124 2413 1324 2143

26. No. Consider Figure 8.3.5.

27. Yes, $(v_1, v_2, v_5, v_4, v_3)$.

29. No. Suppose the graph has a Hamiltonian path. First note that the path must either start or end at either $a$ or $c$. If not, we must use edges $(a, b)$, $(a, d)$, $(b, c)$, and $(c, d)$, which make a cycle. Similarly, the path must start or end at either $j$ or $l$. Suppose that the path starts at $a$.

The path begins with either $(a, b)$ or $(a, d)$. By symmetry, we may assume that the path begins with $(a, d)$.

First suppose that the path ends at $l$. Since the path does not end at $c$, it must include edges $(b, c)$ and $(c, d)$. Also, since the path does not end at $j$, it must include edges $(i, j)$ and $(j, k)$. The path must include either $(b, e)$ or $(b, f)$. If $(b, e)$ is in the path, then $(f, e)$ and $(f, i)$ must be in the path. Since the path ends at $l$, it must contain $(l, k)$. If $(b, f)$ is in the path, then $(f, e)$ and $(e, i)$ must be in the path. Again, since the path ends at $l$, it must contain $(l, k)$. In either case, $h$ must have degree one, which is a contradiction. Therefore the path cannot end at $l$.

Now suppose that the path ends at $j$. Since the path does not end at $c$, it must include edges $(b, c)$ and $(c, d)$. Also, since the path does not end at $l$, it must include edges $(i, l)$ and $(k, l)$. Arguing as in the previous case, we find that either $e$ or $f$ connects to $i$. Since the path ends at $j$, it must include $(j, k)$. Again, $h$ must have degree one, which is a contradiction. Therefore the path cannot end at $j$.

Now suppose that the path begins at $c$. By symmetry, we may assume that the path begins with $(c, d)$. Since the path does not end at $a$, it must include edges $(d, a)$ and $(a, b)$. Now the argument is exactly as in the preceding paragraphs; again a contradiction is reached. The proof is complete.

30. No. We would have to eliminate at least one edge at $f$, at least three edges at $c$, at least one edge at $b$, at least one edge at $i$, at least three edges at $j$, at least three edges at $m$, and at least one edge at $p$ leaving 14 edges. Since there are 16 vertices, a Hamiltonian path would have 15 edges.

32. Yes, $(a, b, c, j, i, m, k, d, e, f, l, g, h)$

33. Yes, $(a, b, c, g, l, m, r, q, p, k, j, f, e, i, n, o, t, s, h, d)$

35. The graph contains a Hamiltonian path for all $m$ and $n$. Start in the upper-left corner. Continue right until reaching the end of this row. Drop down to the next row. Continue left until reaching the end of this row. Drop down to the next row. Now repeat, continue right until reaching the end of this row ... Continue until all vertices have been visited.

36. For all $n$

## Section 8.4

2. 11; $(a, b, c, g)$ 3. 10; $(a, b, c, d, z)$ 5. 10; $(h, f, c, d)$

7. Change line 8 of Algorithm 8.4.1 to

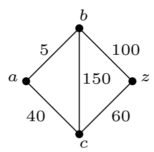
   while $(T \neg= \emptyset)$ {

8. Input: A connected, weighted graph with $n$ vertices in which all weights are positive (if there is no edge between $i$ and $j$, set $w(i, j) = \infty$)

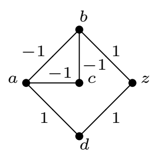
   Output: $\mathrm{dist}(i, j)$, the length of a shortest path from $i$ to $j$ for all $i$ and $j$

```
all_paths(w, n) {
   for j = 1 to n
      for k = 1 to n
         dist(j, k) = w(j, k)
   for i = 1 to n
      for j = 1 to n
         for k = 1 to n
            if (dist(j, i) + dist(i, k) < dist(j, k))
               dist(j, k) = dist(j, i) + dist(i, k)
}
```

10. False




11. False




## Section 8.5

2. Relative to the ordering $a, b, c, d, e, f, g$, the adjacency matrix is

$$\begin{pmatrix} 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

3. Relative to the ordering $a, b, c, d, e$, the adjacency matrix is

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

5. Every entry is one except along the main diagonal, which consists of zeros.

6. If $V_1 = \{a, b\}$ and $V_2 = \{c, d, e\}$ are the vertex sets and the ordering is $a, b, c, d, e$, the adjacency matrix is

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

8. Relative to the orderings $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}$ and $a, b, c, d, e, f, g$, the adjacency matrix is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

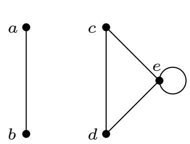
9. Relative to the orderings $x_1, x_2, x_3, x_4$ and $a, b, c, d, e$ the adjacency matrix is

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

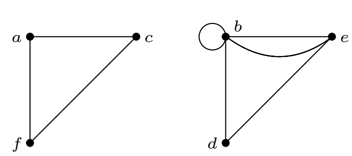
11. $$\begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

12. $$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

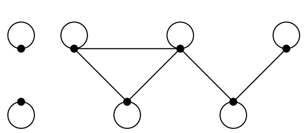
14.




15.




17.



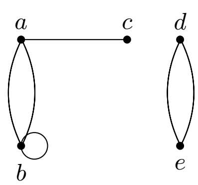
18. (For Exercise 15) The first matrix is relative to the ordering $a, c, f$ and the second is relative to the ordering $b, d, e$.

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 2 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix}$$

20. 93

21. The graph is not connected.

25.




26. The vertex corresponding to the row of zeros is an isolated vertex.

29. Use the fact that

$$\begin{pmatrix} d_{n+1} & a_{n+1} & \cdots & a_{n+1} \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ a_{n+1} & a_{n+1} & \cdots & d_{n+1} \end{pmatrix} = A^{n+1} = \begin{pmatrix} d_n & a_n & \cdots & a_n \\ a_n & d_n & \cdots & a_n \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ a_n & a_n & \cdots & d_n \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}.$$

30. We solve the second-order linear homogeneous recurrence relation (see Exercise 29)

$$a_n = 3a_{n-1} + 4a_{n-2}$$

by the method of Section 7.2.

Solving the equation

$$t^2 - 3t - 4 = 0$$

for $t$, we obtain $t = 4$ and $t = -1$. Thus the solution is of the form

$$a_n = b4^n + d(-1)^n.$$

The initial conditions give the equations

$$\begin{aligned} 1 &= a_1 = 4b - d \\ 3 &= a_2 = 16b + d. \end{aligned}$$

Solving for $b$ and $d$, we obtain $b = 1/5$ and $d = -1/5$. Therefore

$$a_n = \frac{4^n}{5} - \frac{(-1)^n}{5} = \frac{1}{5}[4^n + (-1)^{n+1}].$$

## Section 8.6

2. Relative to the vertex orderings $a, b, c, d, e, f$ for $G_1$, and $4, 2, 1, 3, 6, 5$ for $G_2$, the adjacency matrices of $G_1$ and $G_2$ are equal.

3. Relative to the vertex orderings $a, b, c, d, e$ for $G_1$, and $3, 4, 1, 5, 2$ for $G_2$, the adjacency matrices of $G_1$ and $G_2$ are equal.

5. Relative to the vertex orderings
$$a, b, c, d, e, f, g, h, i, j$$
for $G_1$ of Exercise 4, and
$$1, 2, 3, 4, 7, 6, 10, 9, 5, 8$$
for the given graph, the adjacency matrices are equal.

6. Replacing the vertex labels in the graph $G_1$ in Exercise 4 with the sets in the table

| *Original Vertex* | *Set* |
|---|---|
| $a$ | $\{1, 2\}$ |
| $b$ | $\{3, 5\}$ |
| $c$ | $\{1, 4\}$ |
| $d$ | $\{2, 5\}$ |
| $e$ | $\{3, 4\}$ |
| $f$ | $\{4, 5\}$ |
| $g$ | $\{2, 4\}$ |
| $h$ | $\{2, 3\}$ |
| $i$ | $\{1, 3\}$ |
| $j$ | $\{1, 5\}$ |

yields the graph $G_1$ in Exercise 4.

8. Since $G_1$ has eight edges and $G_2$ has nine edges, the graphs are not isomorphic.

9. The graphs are not isomorphic. $G_2$ has a vertex of degree 4, but $G_1$ has no vertex of degree 4.

11. The graphs are isomorphic. Relative to the vertex orderings $a, b, c, d, e, f, g, h, i, j, k, l$ for $G_1$, and $1, 5, 6, 2, 3, 7, 8, 4, 9, 10, 11, 12$ for $G_2$, the adjacency matrices of $G_1$ and $G_2$ are equal.

12. The graphs are not isomorphic. $G_1$ has a vertex of degree 2, but $G_2$ does not.

14. The graphs are not isomorphic. $G_1$ has two simple cycles of length 3, but $G_2$ has only one simple cycle of length 3 (see also Exercise 19).

15. The graphs are isomorphic. Relative to the vertex orderings $a, b, c, d, e, f, g, h$ for $G_1$, and $5, 3, 1, 7, 4, 6, 2, 8$ for $G_2$, the adjacency matrices of $G_1$ and $G_2$ are equal.

16. Extend the definition in Example 8.6.3 as follows: $f(v) = v_1 v_2 \ldots v_k$, where $v_i$ is the $i$th coordinate determined as the members of a $t_i$ Gray code. Note that if $(v, w)$ is an edge in $M$,

the strings $v_1 v_2 \ldots v_k$ and $w_1 w_2 \ldots w_k$ will differ in exactly one bit. So, $(v, w)$ is an edge of the $(t_1 + t_2 + \cdots + t_k)$-cube. Define $g$ on the edges of $M$ by

$$g((v, w)) = (v_1 v_2 \ldots v_k w_1 w_2 \ldots w_k).$$

$f$ and $g$ define an isomorphism from $M$ onto the subgraph $(V, E)$ of the $(t_1 + t_2 + \cdots + t_k)$-cube where

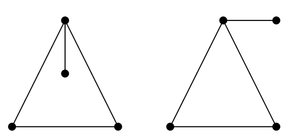
$$\begin{aligned} V &= \{f(v) \mid v \text{ is a vertex in } M\}, \\ E &= \{f(e) \mid e \text{ is an edge in } M\}. \end{aligned}$$

18. Suppose that $G_1$ and $G_2$ are isomorphic. We use the notation of Definition 8.6.1. Suppose that $G_1$ has $n$ vertices $v_1, \ldots, v_n$ of degree $k$ and that $G_2$ has $m$ vertices of degree $k$. By Example 8.6.8, $f(v_1), \ldots, f(v_n)$ each have degree $k$ in $G_2$. Therefore $m \geq n$. By symmetry, $m \leq n$. Thus $m = n$.

19. We use the notation of Definition 8.6.1. Suppose that $G_1$ is connected. We must show that $G_2$ is connected. Let $v'$ and $w'$ be distinct vertices in $G_2$. Then there exist vertices $v$ and $w$ in $G_1$ with $f(v) = v'$ and $f(w) = w'$. Since $G_1$ is connected, there exists a path $(v_0, v_1, \ldots, v_n)$ in $G_1$ with $v_0 = v$ and $v_n = w$. Now $(f(v_0), f(v_1), \ldots, f(v_n))$ is a path in $G_2$ from $v'$ to $w'$. Therefore $G_2$ is connected.

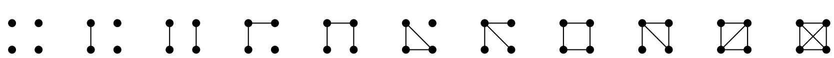
21. Let $(v, w)$ be an edge in $G_1$ with $\delta(v) = i$ and $\delta(w) = j$. Example 8.6.8 shows that $\delta(f(v)) = i$ and $\delta(f(w)) = j$. Now the edge $(f(v), f(w))$ has the desired property in $G_2$.

24. Not an invariant

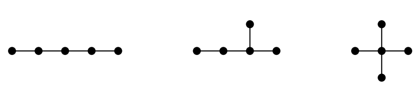


25. Invariant

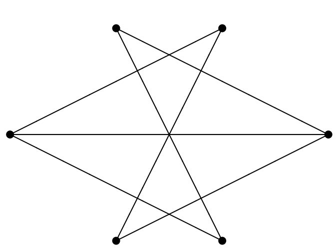
27.



28.



32.



33. Suppose that $G$ is not connected. Let $C$ be a component of $G$ and let $V_1$ be the set of vertices in $G$ that belong to $C$. Let $V_2$ be the set of vertices in $G$ not in $V_1$. In $\overline{G}$, for every $v_1 \in V_1$ and $v_2 \in V_2$, there is an edge $e$ incident on $v_1$ and $v_2$. Thus, in $\overline{G}$ there is a path from $v$ to $w$ if $v \in V_1$ and $w \in V_2$. Suppose that $v$ and $w$ are in $V_1$. Choose $x \in V_2$. Then $(v, x, w)$ is a path from $v$ to $w$. Similarly if $v$ and $w$ are in $V_2$, there is a path from $v$ to $w$. Thus $\overline{G}$ is connected.

35. Suppose that $G_1$ and $G_2$ are isomorphic. We use the notation of Definition 8.6.1. We construct an isomorphism for $\overline{G_1}$ and $\overline{G_2}$. The function $f$ is unchanged. Let $(v, w)$ be an edge in $\overline{G_1}$. Set $g((v, w)) = (f(v), f(w))$.

    It can be verified that the functions $f$ and $g$ provide an isomorphism of $\overline{G_1}$ and $\overline{G_2}$.

    If $\overline{G_1}$ and $\overline{G_2}$ are isomorphic, by the preceding result, $\overline{\overline{G_1}} = G_1$ and $\overline{\overline{G_1}} = G_2$ are isomorphic.

36. Yes

39. $f(1) = w,\ f(2) = x,\ f(3) = y,\ f(4) = z,\ f(5) = y,\ f(6) = x$

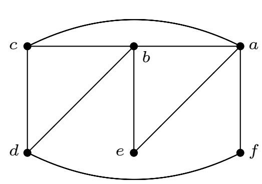
40. $f(1) = a,\ f(2) = b,\ f(3) = c,\ f(4) = d,\ f(5) = c,\ f(6) = b$

42. $f(a) = 1,\ f(b) = 2,\ f(c) = 3,\ f(d) = 4,\ f(e) = 5,\ f(f) = 3,\ f(g) = 2$

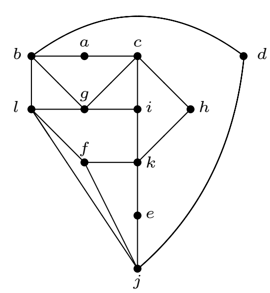
43. See [Hell].

## Section 8.7

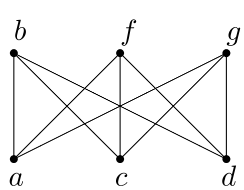
2.



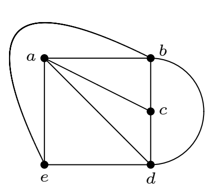

3.



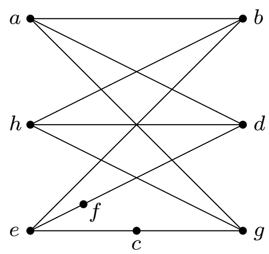

5. Remove $(g, e)$ and $(a, c)$ to obtain a graph homeomorphic to




7. Planar



8. Not planar. The following graph is homeomorphic to $K_{3,3}$.




11. Let $G$ be a graph having four or fewer vertices. By Exercise 10, the planarity of $G$ is not affected by deleting

 loops or parallel edges; so we can assume that $G$ has neither loops nor parallel edges. Now $G$ is a subgraph of $K_4$ and, since $K_4$ is planar, so is $G$.

13. Since every cycle has at least three edges, each face is bounded by at least three edges. Thus the number of edges that bound faces is at least $3f$. In a planar graph, each edge belongs to at most two bounding cycles. Therefore $2e \geq 3f = 3(e - v + 2)$. Thus $3v - 6 \geq e$.

14. $K_{3,3}$

16. Suppose that $G$ and $\overline{G}$ are both planar. Let $v$ denote the number of vertices in $G$. Let $e$ (respectively, $\overline{e}$) denote the number of edges in $G$ (respectively, $\overline{G}$). If either $G$ or $\overline{G}$ is not connected, add just enough edges, preserving planarity, to connect it. Let the connected graphs so obtained be denoted $G^*$ (with $e^*$ edges) and $\overline{G^*}$ with $\overline{e^*}$ edges). Using Exercise 13, we obtain

$$\frac{v(v-1)}{2} = e + \overline{e} \leq e^* + \overline{e^*} \leq 2(3v - 6).$$

 Thus

$$v^2 - 13v + 24 \leq 0.$$

 The roots of the equation obtained by replacing $\leq$ by $=$ are

$$x = \frac{13 \pm \sqrt{73}}{2},$$

 so

$$v \leq \frac{13 + \sqrt{73}}{2} < 11.$$

17. See *Amer. Math. Mo.*, April 1983, pages 287–288.

19. Pick a city in each country. Draw a line through the common border between two cities in countries sharing a common border. This can be done with no lines crossing.

20. Color $D$, say, red. Now $B$ and $C$ must be different colors and different from red.

21. $A$–red, $B$–green, $C$–blue, $D$–red, $E$–green, $F$–blue, $G$–green

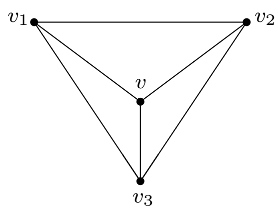
23. Color $L$ red. Now $G$ needs a different color–say blue. Now $K$ needs a color different from $L$ and $G$–say yellow. Now $J$ needs a fourth color.

24. $A$–blue, $B$–green, $C$–red, $D$–yellow, $E$–green, $F$–red, $G$–yellow, $H$–green, $I$–yellow, $J$–green, $K$–blue, $L$–red

26. Suppose that $G'$ can be colored with $n$ colors. If we eliminate edges from $G'$ to obtain $G$, $G$ is colored with $n$ colors.

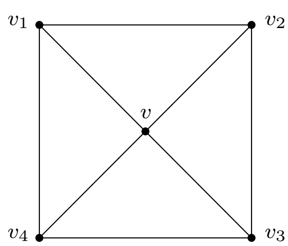
27. Each face is bounded by three edges and each edge is in a boundary for two faces.

29. Suppose that $G$ has a vertex of degree 3. Then, we find the configuration




Consider the map $G'$ obtained from $G$ by removing vertex $v$ and the three edges incident on $v$. By assumption, $G'$ can be colored with four colors. Now $v_1$, $v_2$, and $v_3$ require at most three colors. Color $v$ with the fourth color. Now $G$ is colored with four colors–a contradiction.

30. If $G$ has a vertex $v$ of degree 4, we find the configuration




Consider the graph $G'$ obtained from $G$ by removing vertex $v$ and the four edges incident on $v$. By assumption, $G'$ can be colored with four colors. Show that if $v_1$, $v_2$, $v_3$, and $v_4$ use three or fewer colors, we get an immediate contradiction.

Suppose that $v_1$, $v_2$, $v_3$, and $v_4$ require four colors and that $v_i$ is colored $C_i$. Consider the subgraph $G'_1$ of $G'$ consisting of all simple paths starting at $v_1$ whose vertices are alternately colored $C_1$ and $C_3$. If $G'_1$ does not include $v_3$, we may change each $C_1$ to $C_3$ and each $C_3$ to $C_1$ in $G'_1$ and produce a coloring of $G'$ with four colors. If this is done, we can then color $G$ with four colors.

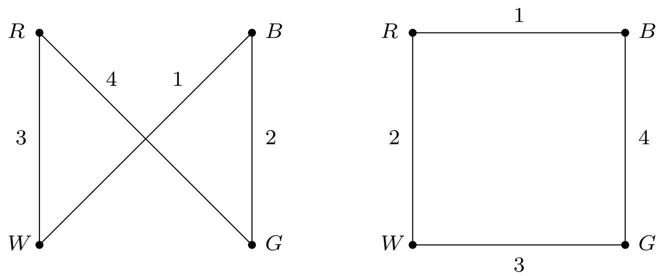
Suppose that $G'_1$ includes $v_3$. Consider the subgraph $G'_2$ of $G'$ consisting of all simple paths starting at $v_2$ whose vertices are alternately colored $C_2$ and $C_4$. Show that $G'_2$ cannot include $v_4$. We may change each $C_2$ to $C_4$ and each $C_4$ to $C_2$ in $G'_2$ and produce a coloring of $G'$ with four colors. If this is done, we can then color $G$ with four colors.

Deduce that $G$ cannot have a vertex of degree 4.

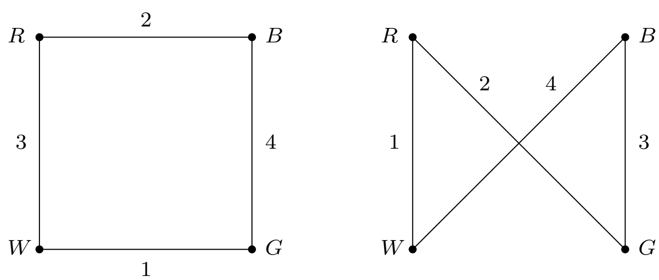
32. Use the methods of Exercise 29–31.

## Section 8.8

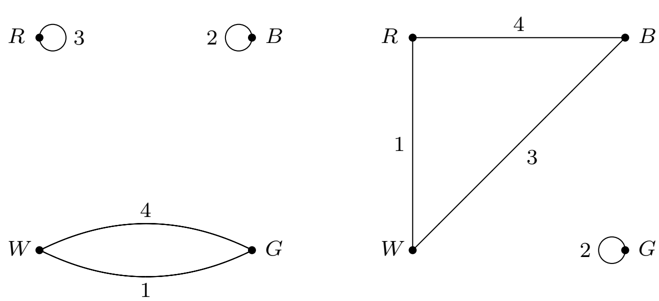
2.




3.

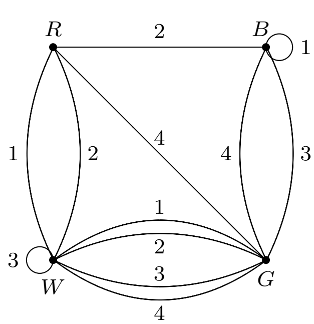



5.

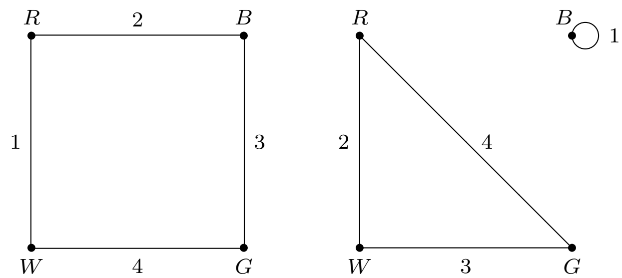



6. There is no solution.

8. (a)

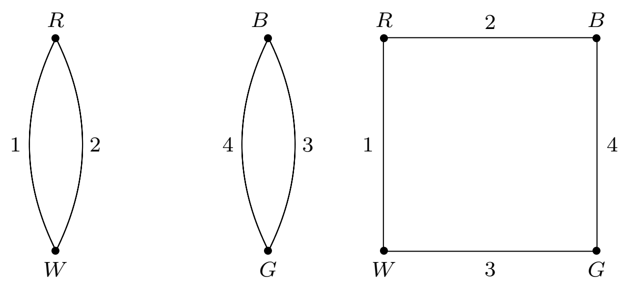


(b)




$G_1$ $G_2$

(c)

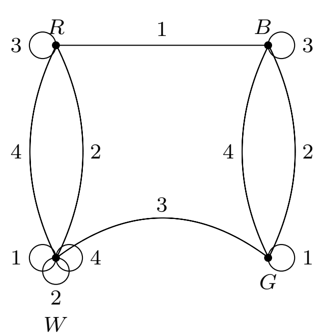



$G_3$ $G_4$

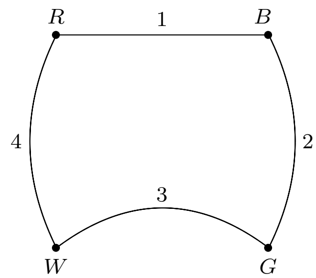
(d) Note that $G_i$ and $G_j$ have edges in common for $i = 1,\ j = 3$; $i = 1,\ j = 4$; $i = 3,\ j = 4$; $i = 2,\ j = 3$; and $i = 2,\ j = 4$. Thus the only solution is $G_1, G_2$.

9. We cannot select the edge incident on $R$ and $B$ for, if we do, there is no way to make the degree of $R = 2$. Similarly, we cannot select the edge incident on $B$ and $G$ for, if we do, there is no way to make the degree of $B = 2$. Since we must have an edge labeled 4, we must select the loop incident on $W$. This means we cannot select any of the edges incident on $W$ and $G$. Now, $G$ cannot have degree 2. Thus, no subgraph satisfies (8.8.1) and (8.8.2).

10. We claim that the following graph has the desired properties.




Note that each cube contains all four colors. The following subgraph satisfies properties (8.8.1) and (8.8.2).




We show that there is no subgraph satisfying (8.8.1) and (8.8.2) that is edge-disjoint from the preceding subgraph. If there is, consider an edge labeled 1. We must choose either the loop labeled 1 on $W$ or the loop labeled 1 on $G$. If we choose the loop labeled 1 on $W$, we cannot add an edge labeled 2. Suppose that we choose the loop labeled 1 on $G$. Now the only edge labeled 4 that we can choose is the loop on $W$. Now we cannot add an edge labeled 2. Therefore there is no subgraph satisfying (8.8.1) and (8.8.2) that is edge-disjoint from the preceding subgraph and, hence, the puzzle has no solution.

11. There are six choices for the top and, having chosen the top, there are four choices for the front for a total of $6 \cdot 4 = 24$ choices.

12. By Exercise 11, there are 24 orientations of one cube. Thus there are $24^4 = 331{,}776$ stackings.

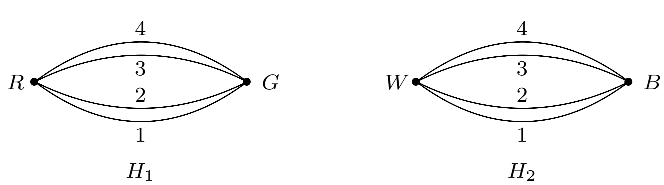
14. Let


be subgraphs of the graph representing the four cubes in the puzzle such that the intersection of the edge sets and the intersection of the vertex sets are empty.

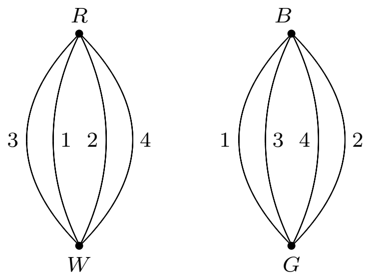
We can use $H_1$ to construct front and back sides of the stack with the front having color $v_1$ and back having color $v_2$. This is possible since the edges incident on $v_1$ and $v_2$ contain all the labels 1, 2, 3, and 4. Similarly, $H_2$ can be used to construct the left and right sides of the stack with the left color $v_3$ and the right color $v_4$.

Any solution is of this form, for if a solution exists, let $v_1$, $v_2$, $v_3$, and $v_4$ be the colors of the front, back, left, and right faces of the solution stack. Then $v_1$ and $v_2$ appear on opposite faces of all four cubes, and $v_3$ and $v_4$ appear on the other opposite faces of all four cubes. Thus $H_1$ and $H_2$ exist in the graph, as shown previously, representing the solution stack.

16.



17.




18. There is no solution.

20. Let $H_1$ and $H_2$ be a solution to the modified version as in, for example, the solution to Exercise 16. We construct subgraphs $G_1$ and $G_2$ as follows. We let the vertex set of $G_1$ be $\{R, B, G, W\}$. We let the edge set of $G_1$ be the set of edges labeled $1, 2$ from $H_1$ and the edges labeled $3, 4$ from $H_2$. We let the vertex set of $G_2$ be $\{R, B, G, W\}$. We let the edge set of $G_2$ be the set of edges labeled $3, 4$ from $H_1$ and the edges labeled $1, 2$ from $H_2$.

21. Yes. See the graph of Exercise 9.

# Chapter 9

# Solutions to Selected Exercises

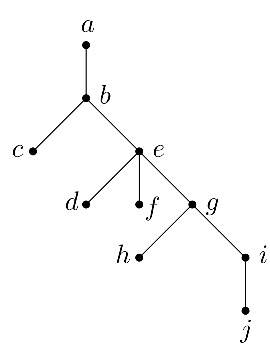
## Section 9.1

2. This graph is a not tree because, if $v$ is the upper-left vertex and $w$ is the bottom, middle vertex, there are two simple paths from $v$ to $w$.

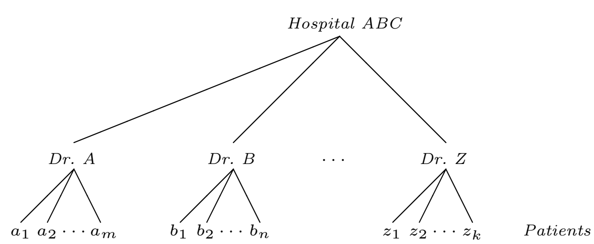
3. This graph is a not tree because, if $v$ is the left, middle vertex and $w$ is the left, bottom vertex, there is no simple path from $v$ to $w$.

5. If either $m$ or $n$, or both, equals 1 6. $n = 1, 2$

9. 4 10. 5



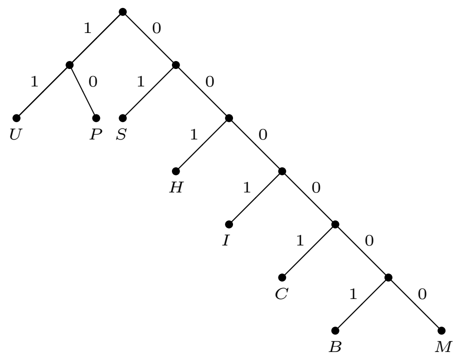

12.

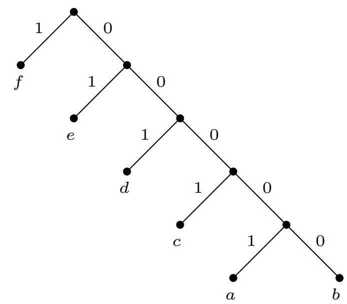



15. *LAP* 16. *DEAL* 19. 010000001111 20. 0111000100111100010

22. Overhead in decoding, memory addressing capability, compatibility with other systems, amount of memory available

23. See G. Williams and R. Meyer, "The Panasonic and Quasar hand-held computers: beginning a new generation of consumer computers," *BYTE*, 6 (January 1981), 34–45.

25. 


28. 


29. The proposed code is ambiguous. For example, 01 could represent $EA$ or $C$.

30. A terminal vertex has degree 1.

31. Since $K_{3,3}$ and $K_5$ contain cycles, a tree cannot contain a subgraph homeomorphic to either; thus, a tree is planar.

33. Consider the tree to be rooted. Color the vertices on even levels one color and those on odd levels another color.

34. $a$–5, $b$–4, $c$–5, $d$–4, $e$–3, $f$–4, $g$–3, $h$–4, $i$–4, $j$–5

36. In this solution, we call a simple path in a tree from the root to a terminal vertex a *drop*. Also, we let $ecc(v)$ denote the eccentricity of the vertex $v$.

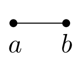
    Let $c$ be a center of a tree $T$. Root $T$ at $c$. Notice that if $ecc(c) = L$, the height of $T$ is $L$.

    We first show that no vertex on level 2 or greater can be a center. For suppose that there is a center $c'$ on level two or greater. Then $ecc(c') = L$. A simple path starting at $c'$ of length $L$ must pass through $c$. But now any simple path starting at the parent of $c'$ has length at most $L - 1$. This contradicts the definition of "center."

    Notice that no vertex different from $c$ on a drop whose length is less than $L$ can be a center. Thus the only possible centers besides $c$ are the children of $c$ which lie on drops of length $L$. It is easy to see that if $c$ has at least two children each lying on a drop of length $L$, then $c$ is the unique center. If $c$ has a unique child $c'$ lying on a drop of length $L$, $c$ and $c'$ are the only centers.

37. In the solution to Exercise 36, we showed that all centers are on level 0 or level 1. Therefore the centers are adjacent.

39. In the following tree, $(a, b)$ and $(a, b, a, b)$ are distinct paths from $a$ to $b$.

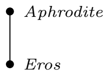



## Section 9.2

2. Aphrodite, Uranus

3. Aphrodite, Kronos, Atlas, Prometheus

5. Zeus, Poseidon, Hades

6. 

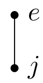

8. Ancestors of $c$: $b, a$. Ancestors of $j$: $e, c, b, a$.

9. Children of $d$: $h, i$. Child of $e$: $j$.

11. Siblings of $f$: $e, g$. Sibling of $h$: $i$.

12. Terminal vertices: $j, f, g, h, i$

14. 

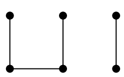

15. 


18. They are siblings.

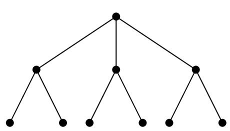
19. It is the root.

20. One is the ancestor of the other.

21. It is a terminal vertex.

23. 

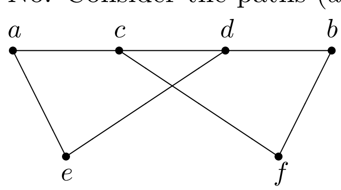
24. No such graph exists. A terminal vertex has degree 1.

26. 

28. In this case, if the edge is $(v, w)$, we would have the cycle $(v, w, v)$.

29. The graph is not a tree since, according to Definition 9.1.1, a tree is a *simple* graph satisfying: If $v$ and $w$ are vertices, there is a unique simple path from $v$ to $w$.

31. $n - m$

32. No. Consider the paths $(a, c, d, b)$ and $(a, e, d, c, f, b)$ in the graph



34. First, suppose that $T$ is a tree. By Theorem 9.2.3b, $T$ is connected. Suppose that for some vertex pair $v, w$, when edge $(v, w)$ is added, at least two cycles are created. Then there must be at least two distinct simple paths from $v$ to $w$ to account for the distinct cycles. But this contradicts the definition of a tree.

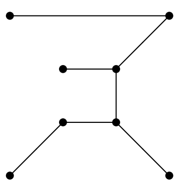
    Now suppose that $T$ is connected and when an edge is added between any two vertices, exactly one cycle is created. It follows that if $v$ and $w$ are vertices in $T$, there is a unique simple path from $v$ to $w$. For if there were no simple path from $v$ to $w$, inserting an edge between $v$ and $w$

would not create a cycle. If there were two or more simple paths between $v$ and $w$, inserting an edge between $v$ and $w$ would create two or more cycles. Thus $T$ is a tree by the definition of tree.

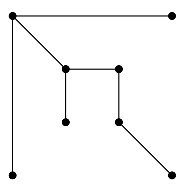
35. Let $v$ be a vertex of degree at least 2 in a tree $G$ and let $P = (v_0, \ldots, v_n)$ be a simple path of maximum length passing through $v$. Since $G$ is a tree, $P$ is not a cycle and, since $v$ has degree at least 2, $v \neq v_0$ and $v \neq v_n$. If removing $v$ and all edges incident on $v$ leaves a connected graph, then there is a simple path, distinct from $P$, from $v_0$ to $v_n$. Since $G$ is a tree, this is impossible. Therefore $v$ is an articulation point.

## Section 9.3

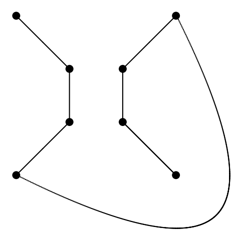
2.



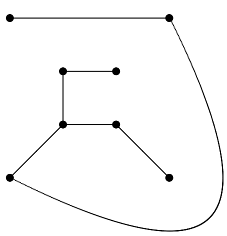
3.



5.

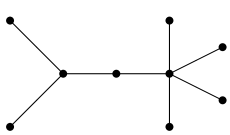


6.



8. The cycle $(a, b, c, d, e, f, g, h, i, j, k, l)$

9.



11.

| | | | |
|---|---|---|---|
| | | × | |
| × | | | |
| | | | × |
| | × | | |

| | | | |
|---|---|---|---|
| | × | | |
| | | | × |
| × | | | |
| | | × | |

12.

| | | | | |
|---|---|---|---|---|
| | | × | | |
| × | | | | |
| | | | × | |
| | × | | | |
| | | | | × |

| | | | | | |
|---|---|---|---|---|---|
| | | | × | | |
| × | | | | | |
| | | | | × | |
| | × | | | | |
| | | | | | × |
| | | × | | | |

14. 10

15. The solution is given as the second part of the solution to Exercise 12.

16. False. Consider $K_4$. A breadth-first search spanning tree will produce a tree whose root has degree 3. Thus it cannot produce the tree $(a, b, c, d)$.

18. If $T$ is a tree, every vertex ordering with the same initial vertex produces the same spanning tree, namely $T$ itself.

19. If $T$ is a tree, every vertex ordering with the same initial vertex produces the same spanning tree, namely $T$ itself.

21. First show that the graph $T$ constructed is a tree. Now use induction on the number of iterations of the loop to show that $T$ contains all of the vertices of $G$.

22. If the edge is not contained in a cycle of $G$

24. Input: A connected graph $G$ with vertices ordered $v_1, \ldots, v_n$; and $d$
Output: $d(v_i)$ = length of a shortest path from $v_1$ to $v_i$

```
short_paths(V, E, d) {
  S = (v1)
  V' = set consisting of v1
  E' = ∅
  d(v1) = 0
  while (true) {
    for each x ∈ S, in order
      for each y ∈ V − V', in order
        if ((x, y) is an edge) {
          add edge (x, y) to E' and y to V'
          d(y) = d(x) + 1
        }
    if (no edges were added)
      return T
    S = children of S ordered consistently with the original vertex ordering
  }
}
```

25. Both algorithms find simple paths from $v$ in increasing order of length.

28. The fundamental cycle matrix relative to the orderings $(a,b,a)$, $(b,d,c,b)$, $(b,c,f,b)$, $(d,e,c,d)$, $(c,f,e,c)$, $(c,e,g,d,c)$, $(c,f,g,d,c)$ and $e_2$, $e_3$, $e_5$, $e_{12}$, $e_{13}$, $e_{10}$, $e_{11}$, $e_1$, $e_4$, $e_6$, $e_7$, $e_8$, $e_9$ is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

29. The fundamental cycle matrix relative to the orderings $(a,b,d,e,a)$, $(a,b,d,e,a)'$, $(b,c,d,b)$, $(d,e,f,d)$, and $e_3$, $e_4$, $e_2$, $e_9$, $e_1$, $e_5$, $e_6$, $e_7$, $e_8$ is

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

31. Modify Algorithm 9.3.7 as follows. Change the line return $T$ to

```
if (|V'| == n)
    return true
else
    return false
```

If the graph is connected, the value true is returned; otherwise, the value false is returned.

32. Modify Algorithm 9.3.10 as follows. Change the line return true to

print solution

Delete the line return false.

35. *dfs_track_parent*$(V, E, parent)$ {
```
    // set v1's parent to 0 to indicate that v1 has no parent
    parent(v1) = 0
    V' = {v1}
    E' = ∅
    w = v1
    while (true) {
        while (there is an edge (w, v) that when added to T does not create a cycle in T) {
            choose the edge (w, vk) with minimum k that when added to T
                does not create a cycle in T
            add (w, vk) to E'
            parent(vk) = w
            add vk to V'
            w = vk
```

```
            }
            if (w == v_1)
                return T
            w = parent of w in T // backtrack
        }
    }
```

36. *print_parents*$(V, parent)$ {

```
        for each v ∈ V
            println(v, parent(v))
    }
```

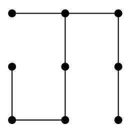
38. The solution array $x$ contains 0's and 1's. If $x[k] = 0$, $k$ is not included in the subset. If $x[k] = 1$, $k$ is included in the subset.

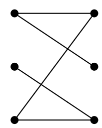
```
    all_subsets(n) {
        rall_subsets(n, 1)
    }

    rall_subsets(n, k) {
        for x[k] = 0 to 1
            if (k == n) {
                for i = 1 to n
                    if (x[i] == 1)
                        print(i + " ")
                println()
            }
            else
                rall_subsets(n, k + 1)
    }
```

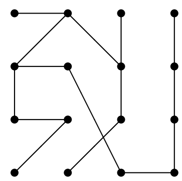
39.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 8 | 7 | 9 | 5 | 2 | 3 | 4 | 6 |
| 6 | 4 | 2 | 8 | 3 | 7 | 1 | 9 | 5 |
| 3 | 5 | 9 | 1 | 4 | 6 | 8 | 7 | 2 |
| 2 | 6 | 4 | 7 | 1 | 3 | 9 | 5 | 8 |
| 5 | 3 | 8 | 2 | 9 | 4 | 7 | 6 | 1 |
| 9 | 7 | 1 | 5 | 6 | 8 | 4 | 2 | 3 |
| 8 | 9 | 5 | 4 | 2 | 1 | 6 | 3 | 7 |
| 7 | 2 | 3 | 6 | 8 | 9 | 5 | 1 | 4 |
| 4 | 1 | 6 | 3 | 7 | 5 | 2 | 8 | 9 |

## Section 9.4

2. 

3. 

5. 

6. Suppose that the start vertex is 1, and that the vertices are added in the order $1, 2, \ldots, n$. When $i$ is 1 and $j$ is 1, the innermost line (line 11) of the nested for loops is executed $n$ times. When $i$ is 2, the innermost line of the nested for loops is executed $n$ times for each of the values $j = 1, 2$. When $i$ is 3, the innermost line of the nested for loops is executed $n$ times for each of the values $j = 1, 2, 3$; and so on. Thus line 11 is executed

$$n + 2n + 3n + \cdots + (n-1)n = n[1 + 2 + 3 + \cdots + (n-1)] = \frac{n(n-1)n}{2} = \Theta(n^3)$$

times. No input requires more than $\Theta(n^3)$ time since the nested for loops take at most $O(n^3)$ time to execute. Therefore the worst-case time is $\Theta(n^3)$.

8. The body of the last for loop executes $n-1$ times the first time, $n-2$ times the second time, and so on. This time dominates, so the worst-case time is

$$(n-1) + (n-2) + \cdots 1 = \frac{(n-1)n}{2} = \Theta(n^2).$$

9. The argument is similar to the proof of Theorem 9.4.5.

11. Yes

12. Suppose that the weight of each edge in $K_n$ is equal to 2. Suppose that some algorithm does not examine edge $e$. Let $T$ denote the minimal spanning tree output by the algorithm. If $e$ is in $T$, alter the input by changing the weight of $e$ to 3. If $e$ is not in $T$, alter the input by changing the weight of $e$ to 1. Rerun the algorithm. Notice that since the algorithm does not examine $e$, it will still output $T$. However, for the modified input, $T$ is not a minimal spanning tree. This is a contradiction. Therefore every minimal spanning tree algorithm examines every edge in $K_n$.

15. True

17. The proof is similar to the proof of Theorem 9.4.5. Let $G_i$ be the graph produced at the $i$th iteration. Use induction to show that $G_i$ contains a minimal spanning tree.

18. In Algorithm 9.4.3, change $\infty$ in line 6 to $-\infty$ and change $<$ to $>$ in line 10.

21. (For Exercise 1) If we break ties by picking the smallest vertices, Kruskal's Algorithm picks, successively, $(2,3)$, $(3,5)$, $(3,4)$, $(1,2)$.

22. Argue as in the proof of Theorem 9.4.5.

24. The algorithm picks one 10-cent stamp and six 1-cent stamps to make 16 cents postage, but two 8-cent stamps is optimal.

25. We use induction on $n$ to show that the greedy solution and any optimal solution to the $n$-cent problem are identical. The statement is clearly true for $n = 1, 2, 3, 4, 5, 25$.

    Suppose that $5 < n < 25$. Let $S$ be an optimal solution to the $n$-cent problem. We must use a 5-cent stamp; for otherwise, we could replace five 1-cent stamps with one 5-cent stamp. Now $S$ with a 5-cent stamp removed is an optimal solution to the $(n-5)$-cent problem; for otherwise, an optimal solution to the $(n-5)$-cent problem together with a 5-cent stamp would be smaller than $S$. By the inductive assumption, $S$, with a 5-cent stamp removed, is the greedy solution. Therefore $S$ is the greedy solution.

    Suppose that $n > 25$. Let $S$ be an optimal solution to the $n$-cent problem. We must use a 25-cent stamp since we can make at most 24 cents postage optimally using only 5-cent and 1-cent stamps. Now $S$ with a 25-cent stamp removed is an optimal solution to the $(n-25)$-cent problem. By the inductive assumption, it is the greedy solution. Therefore $S$ is the greedy solution.

26. $a_1 = 11,\ a_2 = 5$. For $n = 15$, the greedy method gives 11, 1, 1, 1, 1, but 5, 5, 5 is better.

28. The set $\{1, 5, 11\}$ shows that the condition is not sufficient. For $n = 15$, the greedy algorithm gives one 11-cent stamp and four 1-cent stamps, but three 5-cent stamps is optimal.

    The set $\{1, 5, 10, 20, 25, 40\}$ shows that the condition is not necessary. (The example is due to Stephen B. Maurer, *Amer. Math. Mo.*, 101 (5), 419.) The greedy algorithm is optimal for these denominations; however, the condition fails for $i = 5$: $25 \geq 2 \cdot 20 - 10$ is false.

    We can use induction to prove that the greedy algorithm is optimal for the set $\{1, 5, 10, 20, 25, 40\}$. We verify directly the cases $1 \leq n \leq 214$. Now suppose that $n > 214$. Let $S$ be an optimal solution for $n$. We claim that $S$ contains a 40-cent stamp. If not, $S$ contains at most four 1-cent stamps (since five 1-cent stamps could be replaced by one 5-cent stamp). For the same reason, $S$ contains at most one 5-cent stamp, at most one 10-cent stamp, at most one 20-cent stamp, and at most seven 25-cent stamps. But now $S$ can make at most

    $$4 \cdot 1 + 1 \cdot 5 + 1 \cdot 10 + 1 \cdot 20 + 7 \cdot 25 = 214$$

    cents postage. This contradiction shows that $S$ contains a 40-cent stamp.

    Now let $G_n$ be the greedy solution for $n$-cents postage, and let $S'$ be $S$ with one 40-cent stamp removed. Then $S'$ is optimal for $(n-40)$-cents postage. By the inductive assumption, $|G_{n-40}| = |S'|$. Therefore

    $$|G_n| = 1 + |G_{n-40}| = 1 + |S'| = |S|,$$

    and the greedy solution is optimal for $n$-cents postage.

30. Let $a_1 = 1$, $a_2 = 5$, and $a_3 = 6$. The greedy algorithm is optimal for $n = 1, \ldots, 9$, but not optimal for $n = 10$.

31. There might be a non-greedy solution for $n$ that does not use a 6-cent stamp. In this case, the greedy algorithm might be optimal for $n - 6$, but not for $n$ (consider $n = 10$).

## Section 9.5

2. $2^{64}$, which has $1 + \lfloor \log_{10} 2^{64} \rfloor = 20$ digits.

3. $1/2^{64} \approx 5.4 \times 10^{-20}$

6. Input: *root*, the root of an $n$-vertex binary search tree; *key*, a value to find; and $n$
   Output: The vertex containing *key*, or *null* if *key* is not in the tree

```
bst_search(root, n, key) {
  ptr = root
  while (ptr ¬= null)
    if (ptr contains key)
      return ptr
    else if (ptr contains a value greater than key)
      ptr = left child of ptr
    else
      ptr = right child of ptr
  return null
}
```

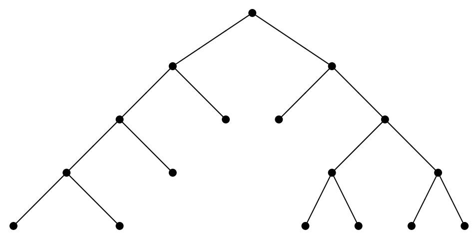
7. Input: $s_1, \ldots, s_n, n$
   Output: A binary search tree $T$ of minimum height that stores $s_1, \ldots, s_n$

```
optimal_bst(s, n) {
  sort s_1, ..., s_n
  return o_bst(s, 1, n)
}
```

```
o_bst(s, i, j) {
  if (i > j)
    return null
  m = ⌊(i + j)/2⌋
  T' = optimal_bst(s, i, m − 1)
  T'' = optimal_bst(s, m + 1, j)
  let T be the tree whose root contains s_m
  let the left subtree of T be T'
  let the right subtree of T be T''
  return T
}
```

10. There is no such graph. The existence of such a graph would contradict Theorem 9.5.6.

11.



13. Input: An integer $n > 1$
Output: A full binary tree $T$ with $n$ terminal vertices

```
full_binary_tree(n) {
   T = a rooted tree with one vertex
   for i = 1 to n − 1 {
      let v be a terminal vertex
      give v two children
   }
   return T
}
```

14. Input: A word $w$ to insert in a binary search tree $T$
Output: The updated binary search tree $T$

```
bst_recurs(w, T)
   if (T == null) {
      let T be the tree with one vertex, root
      store w in root
      return T
   }
   s = word in T's root
   if (w < s)
      if (T has no left child)
         give T a left child and store w in it
      else {
         left = left child of T
         bst_recurs(w, left)
      }
   else
      if (T has no right child)
         give T a right child and store w in it
      else {
         right = right child of T
         bst_recurs(w, right)
      }
   return T
}
```

16. Input: The root *root* of a nonempty binary tree in which data are stored

Output: true, if the binary tree is a binary search tree; false, if the binary tree is not a binary search tree. If the binary tree is a binary search tree, the algorithm sets *small* to the smallest value in the tree and *large* to the largest value in the tree.

```
is_bst(root, small, large) {
   if (root has no children) {
      small = value of root
      large = value of root
      return true
   }
   lchild = left child of root
   rchild = right child of root
   if (is_bst(lchild, small_left, large_left) ∧ is_bst(rchild, small_right, large_right)) {
      val = value of root
      if (large_left > val ∨ small_right < val)
         return false
      small = small_left
      large = large_right
      return true
   }
   else
      return false
}
```

17. We prove the result using induction on $n$.

BASIS STEP ($n = 1$). In this case, the tree consists of three vertices–the root and its two children. Thus $I = 0$, $E = 2$, and $E = 2 = 0 + 2 \cdot 1 = I + 2n$.

INDUCTIVE STEP. Assume that the equation is true for $n$. Let $T$ be a tree with $n + 1$ internal vertices. Let $T'$ be the tree obtained from $T$ by deleting two sibling terminal vertices and the edges incident on them. Let $p$ denote the (former) parent of the deleted siblings. The resulting tree $T'$ has $n$ internal vertices. Let $I'$ and $E'$ denote the internal and external path lengths for $T'$. By the inductive assumption, $E' = I' + 2n$.

If $len$ denotes the length of the simple path from the root to $p$ in $T$, the external path length in $T$ is obtained from the external path length in $T'$ by adding $2(len + 1)$, to account for the two new paths to the children of $p$, and by subtracting $len$, to account for the loss of the path to the former terminal $p$; thus,

$$E = E' + 2(len + 1) - len = I' + 2n + len + 2.$$

The internal path length in $T'$ is obtained from the internal path length in $T$ by subtracting $len$ to account for the loss of the path to $p$; thus,

$$E = I' + 2n + len + 2 = I - len + 2n + len + 2 = I + 2(n + 1).$$

19. Balanced 20. Not balanced

23. If the balanced trees of heights $h-1$ and $h-2$ with the minimum number of vertices are found, the required tree of height $h$ can be formed by attaching these two trees as right and left subtrees of a new root. Thus $N_h = N_{h-1} + N_{h-2} + 1$.

24. Let $s_h = N_h + 1$. Then

$$s_h = N_h + 1 = 1 + N_{h-1} + N_{h-2} + 1 = s_{h-1} + s_{h-2},$$

by Exercise 23. Now $s_0 = N_0 + 1 = 2$, $s_1 = N_1 + 1 = 3$ (Exercise 22). Thus $\{s_h\}$ satisfies the same recurrence relation as the Fibonacci sequence. Since $s_0 = f_3$ and $s_1 = f_4$, it follows that $s_h = f_{h+3},\ h \geq 0$. Therefore $N_h = s_h - 1 = f_{h+3} - 1$.

26. We prove that $n < 2^{h+1}$ using induction on $n$. Taking lg of both sides gives the desired result. We omit the Basis Step ($n = 1$).

Assume that the result is true for binary trees with less than $n$ vertices. Let $T$ be an $n$-vertex binary tree. Let $n_L$ be the number of vertices in $T$'s left subtree, and let $n_R$ be the number of vertices in $T$'s right subtree. Let $h_L$ be the height of $T$'s left subtree, and let $h_R$ be the height of $T$'s right subtree. Note that $1 + h_L \leq h$ and $1 + h_R \leq h$. By the inductive assumption, $n_L < 2^{h_L+1}$ and $n_R < 2^{h_R+1}$. Now

$$n = 1 + n_L + n_R < 1 + 2^{h_L+1} + 2^{h_R+1} \leq 1 + 2^h + 2^h = 1 + 2 \cdot 2^h = 1 + 2^{h+1}.$$

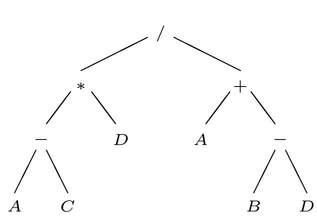
## Section 9.6

2. Preorder: $ABCDEF$
   Inorder: $CBEFDA$
   Postorder: $CFEDBA$

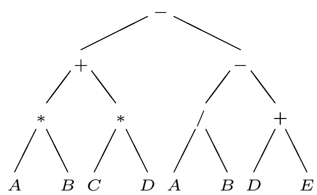
3. Preorder: $ABHIKLMJCDEFG$
   Inorder: $ILKMHJBADFEGC$
   Postorder: $LMKIJHBFGEDCA$

5. Preorder: $ABCDEFG$
   Inorder: $DCBAEFG$
   Postorder: $DCBGFEA$

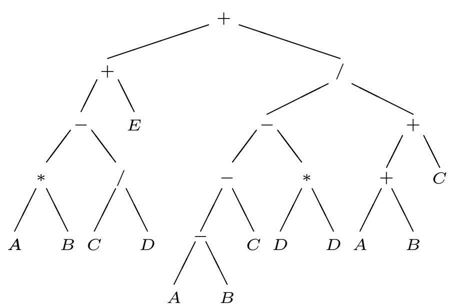
7. Prefix: $/{*}{-}ACD{+}A{-}BD$
   Postfix: $AC{-}D{*}ABD{-}{+}/$



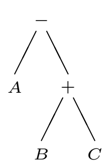
8. Prefix: $-+*AB*CD-/AB+DE$
Postfix: $AB*CD*+AB/DE+--$



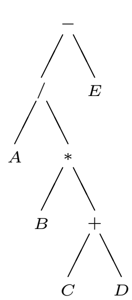
10. Prefix: $++-*AB/CDE/---ABC*DD++ABC$
Postfix: $AB*CD/-E+AB-C-DD*-AB+C+/+$



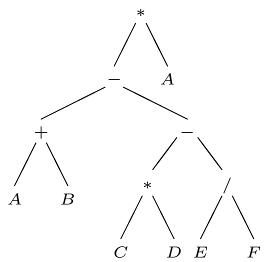
12. Prefix: $-A+BC$
Usual Infix: $A-(B+C)$
Parened Infix: $(A-(B+C))$



13. Prefix: $-/A*B+CDE$
Usual Infix: $A/(B*(C+D))-E$
Parened Infix: $((A/(B*(C+D)))-E)$

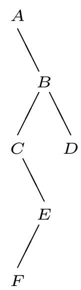


15. Prefix: $*-+AB-*CD/EFA$
Usual Infix: $(A+B-(C*D-E/F))*A$
Parened Infix: $(((A+B)-((C*D)-(E/F)))*A)$




17. 0 18. $-16$ 20. 16 21. $-6$

23. The tree is




Because of the preorder listing, $A$ is the root. If $A$ had a left child, the inorder listing would not begin with $A$. Since $A$ has no left child, the preorder listing tells us that the right child of $A$ is $B$. The argument that the tree is correct continues in this way.

24. Input: $pr$, the preorder list, and $in$, the inorder list
Output: $root$, the root of the binary tree with the given preorder and inorder lists

```
make_tree(pr, in) {
   if (|pr| == 0)
      return null
   ch = first character in pr
   create a vertex v
   store ch in v
   root = v
   choose strings st1 and st2 such that in = st1 + ch + st2 // + is concatenation
   let pr′ be the substring of pr obtained by omitting ch
   choose strings st1′ and st2′ such that pr′ = st1 + st2′, where st1′ (respectively, st2′) is a
      permutation of st1 (respectively, st2)
   left subtree of root = make_tree(st1′, st1)
   right subtree of root = make_tree(st2′, st2)
   return root
}
```

26. Not necessarily. Consider $P_1 = ABCDEF$ and $P_2 = DBCAEF$.

27. Input: $pt$, the root of a binary tree
Output: contents of the terminal vertices from left to right

```
print_terminals(pt) {
   if (pt == null)
      return
   if (pt is a terminal) {
      print contents of pt
      return
   }
   left = left child of pt
   print_terminals(left)
   right = right child of pt
   print_terminals(right)
}
```

29. Input: $pt$, the root of a binary tree
Output: initialize each vertex to the number of its descendants

```
descendants(pt) {
   if (pt == null)
      return
   numb_desc = 0
   left = left child of pt
   if (left ¬= null) {
      descendants(left)
      numb_desc = 1 + contents(left)
   }
   right = right child of pt
   if (right ¬= null) {
      descendants(right)
      numb_desc = numb_desc + 1 + contents(right)
   }
   contents(pt) = numb_desc
}
```

30. Input: $pt$, the root of a binary tree
Output: the number of terminal nodes in $pt$

```
terminals(pt) {
   if (pt == null) {
      return 0
   left = left child of pt
   right = right child of pt
   if (left == null ∧ right == null)
      return 1
   return terminals(left) + terminals(right)
}
```

33. Input: $pt$, the root of a binary tree that represents an expression
 Output: the fully parenthesized infix form of the expression

```
print_expression(pt)
   if (pt == null)
      return
   if (pt is a terminal) {
      print(contents(pt))
      return
   }
   print("(")
   left = left child of pt
   print_expression(left)
   print(contents(pt))
   right = right child of pt
   print_expression(right)
   print(")")
}
```

34. Input: $pt$, the root of a Huffman coding tree, and a string $\alpha$
 Output: The characters and their codes. Each code is prefixed by $\alpha$. To print just the codes, invoke this procedure with $\alpha$ set to the null string.

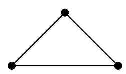
```
huffman(pt, α)
   if (pt is a terminal) {
      print(character stored in pt)
      print(α)
      return
   }
   left = left child of pt
   huffman(left, α + 1) // + is concatenation
   right = right child of pt
   huffman(right, α + 0)
}
```

36. First, note that any subset of $n-1$ vertices is a vertex cover. Second, note that any subset $V'$ of $n-2$ edges is *not* a vertex cover. [If $v$ and $w$ are distinct vertices not in $V'$, then edge $(v, w)$ violates the condition that either $v$ or $w$ is in $V'$.]

37. No. Exercise 36 shows that even if *all* edges are present, $n-1$ vertices suffice for a cover.

39. Let $E'$ be an edge disjoint set for $G$, and let $V'$ be a vertex cover of $G$. We define a function $f$ from $E'$ to $V'$ in the following way: Let $e = (v, w) \in E'$. Then either $v$ or $w$ is in $V'$. Choose one of $v$ or $w$ that is in $V'$, but not both, and let $f(e)$ denote the chosen vertex. The function $f$ is one-to-one because the set $E'$ is an edge disjoint set. Therefore $|E'| \leq |V'|$.

40. The graph

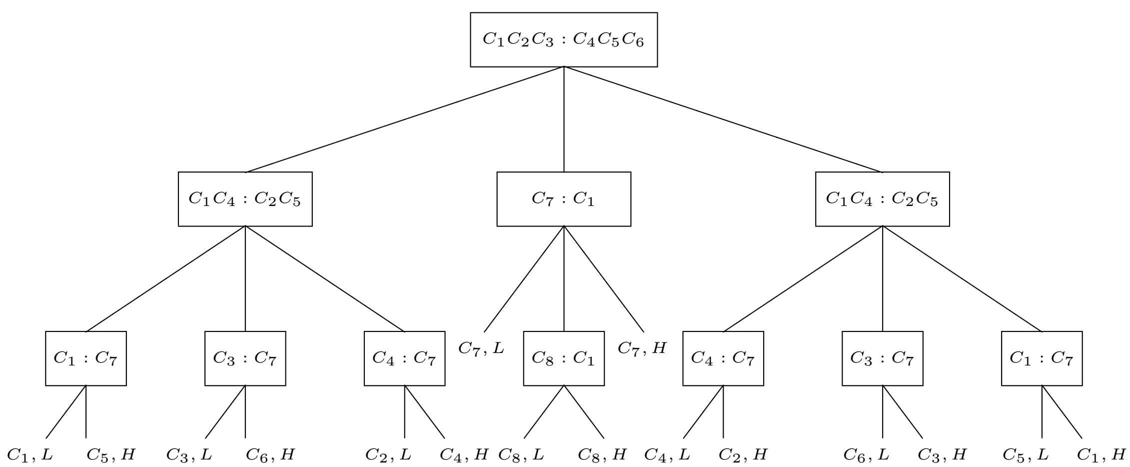


has the desired property since it is impossible to put more than one edge in an edge disjoint set and a single vertex is *not* a vertex cover.

## Section 9.7

2. A tree of height one has at most three terminal nodes. Since four outcomes are possible, the decision tree must have height at least two. Thus at least two weighings are required to solve the problem of Exercise 1.

3. In the following figure, if the coins in the left pan weigh less than the coins in the right pan, we go to the left child. If the coins in the left pan weigh more than the coins in the right pan, we go to the right child. If the coins in the left pan weigh the same as the coins in the right pan, we go to the middle child.



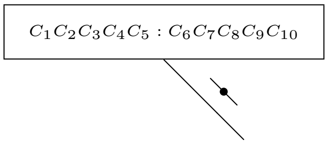

5. If we weigh four coins against four coins and they balance, the problem does *not* reduce to the problem of finding the bad coin from among four coins, but rather to the problem of finding the bad coin from among four coins and *eight good coins*. This latter problem can be solved in at most two weighings.

6. Four weighings are required in the worst case. We prove this result by considering several cases.

   Suppose that we begin by weighing four coins against four coins. If they balance, in two additional weighings, we can account for at most nine outcomes. Since ten outcomes are possible with five coins, we cannot identify the bad coin in at most three weighings in this case.

   Similarly, if we begin by weighing three coins against three coins, two coins against two coins, or one coin against one coin, at least four weighings are required in the worst case.

   Suppose that we begin by weighing five coins against five coins. Consider one of the instances in which they do not balance:




In two more weighings, we can account for at most nine outcomes, but there are ten

$$C_1, L,\ C_2, L,\ C_3, L,\ C_4, L,\ C_5, L,\ C_6, H,\ C_7, H,\ C_8, H,\ C_9, H,\ C_{10}, H.$$

Therefore, if we begin by weighing five coins against five coins, at least four weighings are required in the worst case.

Similarly, if we begin by weighing six coins against six coins, at least four weighings are required in the worst case.

We conclude that the 13-coins puzzle requires at least four weighings in the worst case. In fact, the puzzle can be solved in at most four weighings: Begin by weighing four coins against four coins. If they do not balance, proceed as in the solution to the 12-coins puzzle (see Exercise 4). If they balance, five coins remain. We can identify a bad coin among five in at most three weighings (see Example 9.7.1).

8. If there is an algorithm that solves the puzzle in $k < n$ weighings, the algorithm can be described by a decision tree of height $k$. Every internal vertex of this tree has at most three children; thus, there can be at most $3^k$ terminal vertices. But there are $2((3^n - 3)/2) = 3^n - 3$ possible outcomes and $3^n - 3 > 3^k$ for $n \geq 2$, $k < n$, which is a contradiction.

10. Pick a coin and call it $x$. Set variables $l$ and $s$ to 0. Compare $x$ with each of the other $n - 1$ coins. In comparing $x$ with a coin $y$, if $x$ is heavier than $y$, increment $l$. If $x$ weighs the same as $y$, increment $s$. If $l \neq 0$, the number of bad coins is equal to $l$. If $l = 0$, the number of bad coins is equal to $s + 1$.

11. Input: $a_1, a_2, a_3, a_4$
Output: $a_1, a_2, a_3, a_4$ (in increasing order)

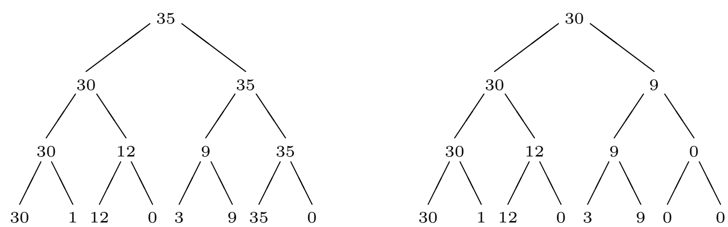
*sort_4*$(a_1, a_2, a_3, a_4)$
  // sort $a_1$ and $a_2$
  if $(a_1 > a_2)$
    *swap*$(a_1, a_2)$
  // sort $a_3$ and $a_4$
  if $(a_3 > a_4)$
    *swap*$(a_3, a_4)$
  // find largest
  if $(a_2 > a_4)$
    *swap*$(a_2, a_4)$
  // find smallest
  if $(a_1 > a_3)$
    *swap*$(a_1, a_3)$
  // sort $a_2$ and $a_3$
  if $(a_2 > a_3)$

$swap(a_2, a_3)$
}

13. There are $6! = 720$ possible outcomes to the problem of sorting six items. To accommodate 720 vertices, we must have a tree of height at least 10 since $2^9 < 720 < 2^{10}$. Thus we need 10 comparisons in the worst case.

    To sort six items using at most 10 comparisons, we first sort five items using at optimal sort (see Exercise 12). This requires at most seven comparisons. We then find the correct position for the sixth item using binary search. This last step requires at most three comparisons.

15.




16. $2^k - 1$

18. $k$

## Section 9.8

2. Not isomorphic. Tree $T_2$ has a vertex of degree 4 ($w_4$), but $T_1$ has no vertex of degree 4.

3. Isomorphic. $f(v_1) = w_3,\ f(v_2) = w_5,\ f(v_3) = w_6,\ f(v_4) = w_2,\ f(v_5) = w_1,\ f(v_6) = w_4$.

5. Not isomorphic. Vertex $v_{10}$ in $T_1$ must be mapped to vertex $w_4$ in $T_2$ since these are the only vertices of degree 4. The vertices adjacent to $v_{10}$ have degree 1, 1, 2, 3, while the vertices adjacent to $w_4$ have degree 1, 2, 3, 2.

6. Isomorphic. $f(v_1) = w_7,\ f(v_2) = w_4,\ f(v_3) = w_6,\ f(v_4) = w_{10},\ f(v_5) = w_3,\ f(v_6) = w_2,\ f(v_7) = w_9,\ f(v_8) = w_{11},\ f(v_9) = w_1,\ f(v_{10}) = w_8,\ f(v_{11}) = w_5,\ f(v_{12}) = w_{12}$.

8. Not isomorphic as rooted trees. The root of $T_1$ has degree 3, but the root of $T_1$ has degree 1.

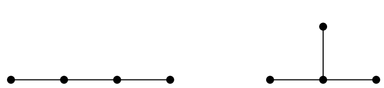
    They are isomorphic as free trees (see the solution to Exercise 3).

9. Isomorphic. $f(v_i) = w_i,\ i = 1, \ldots, 5$. Also, they are isomorphic as free trees.

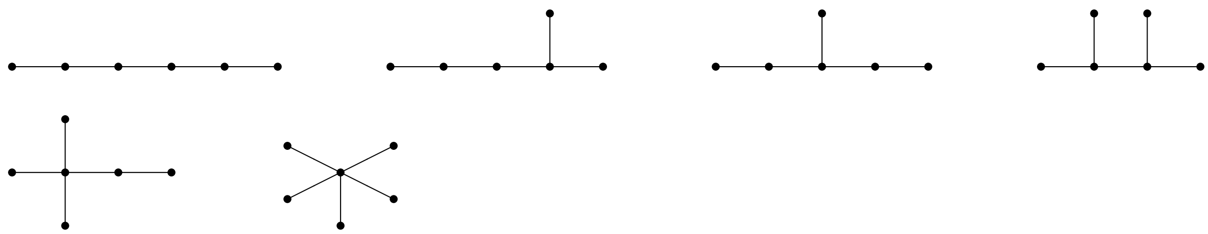
11. Isomorphic. $f(v_i) = w_i,\ i = 1, \ldots, 6$. Also, they are isomorphic as rooted trees and as free trees.

12. Not isomorphic. The root of $T_1$ has no right child, but the root of $T_2$ has a right child. They are not isomorphic as rooted trees, but they are isomorphic as free trees.

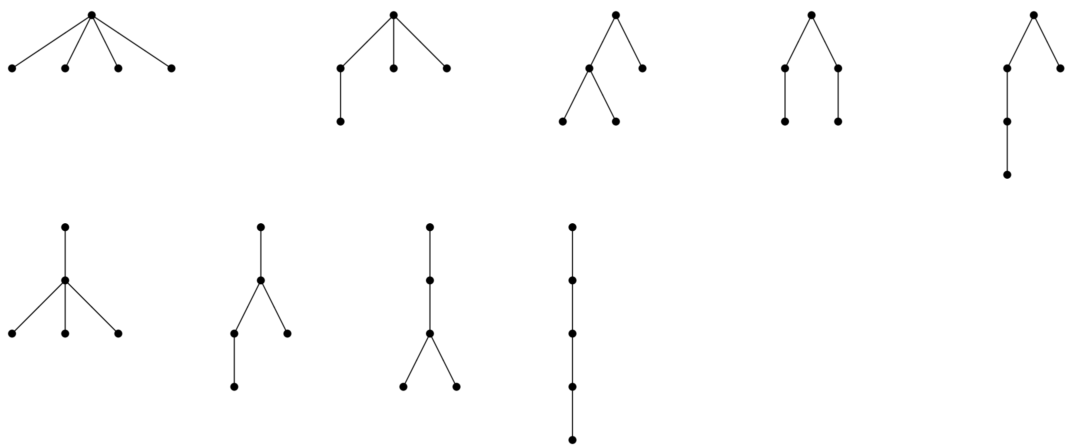
14.



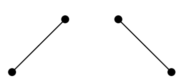
15.



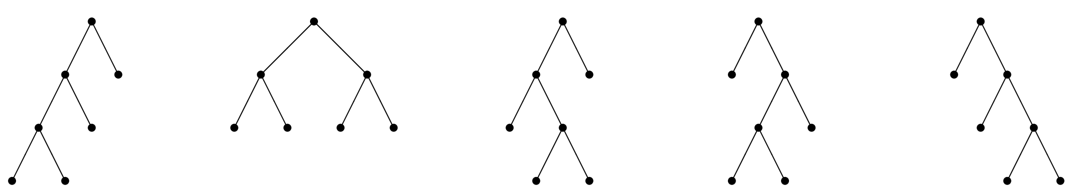
17.



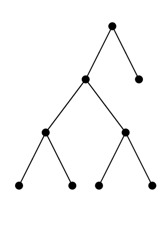
18.

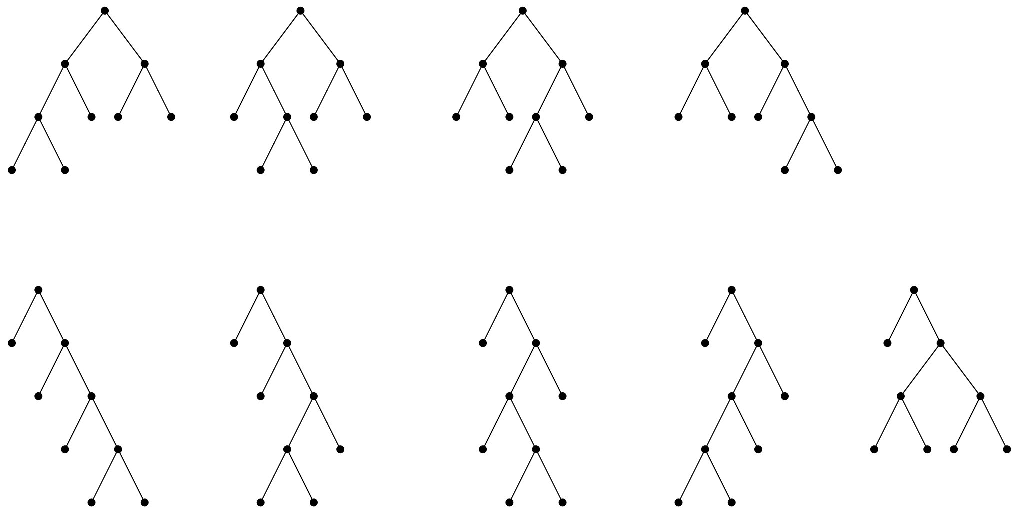


20.

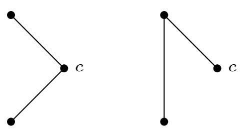


21.

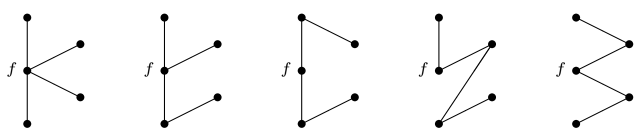




23. For Exercise 9, there are 10 spanning trees obtained by replacing the left triangle by




and by replacing the leftmost figure by




24. Let $b_k$ denote the number of comparisons when two $k$-vertex isomorphic binary trees are input to Algorithm 9.8.13. We use induction on $k$ to show that

$$b_k = 6k + 2. \tag{9.1}$$

If $k = 0$, the trees are empty. In this case, there are two comparisons at line 1 after which the procedure returns. Thus (9.1) is correct for $k = 0$.

Assume that

$$b_i = 6i + 2$$

for $i < k$. There are four comparisons at lines 1 and 3. Let $L$ denote the number of vertices in the left subtree (of either tree) and $R$ denote the number of vertices in the right subtree (of either tree). By the inductive assumption, line 9 requires

$$b_L + b_R = (6L + 2) + (6R + 2)$$

comparisons. Thus the total number of comparisons is

$$4 + 6L + 2 + 6R + 2 = 6(1 + L + R) + 2 = 6k + 2.$$

The inductive step is complete.

26. Input: $n$
Output: an $n$-vertex random binary tree

$rand\_bin\_tree(n)$ {
  if ($n == 0$)
    return null
  let $k$ be a random integer between 0 and $n-1$ inclusive
  $T_1 = rand\_bin\_tree(k)$
  $T_2 = rand\_bin\_tree(n-1-k)$
  let $T$ be the binary tree with left subtree $T_1$ and right subtree $T_2$
  return $T$
}

28. By Exercise 27, there are $C_n$ nonisomorphic full binary trees with $n+1$ terminal vertices. We may choose a terminal vertex to mark in $n+1$ ways. Therefore $|X_2| = (n+1)C_n$.

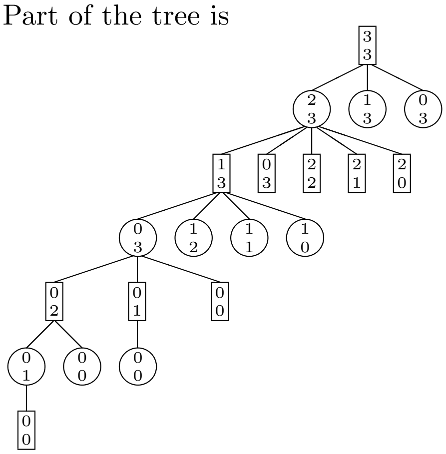
31. Let $T' \in X_2$. Let $t$ denote the marked terminal vertex, let $v$ be $t$'s parent, and let $t'$ be $v$'s sibling. Delete $t$, delete edge $(v, t')$, and replace $v$ by $t'$. If $T$ is the resulting tree, $T' \in X_T$.

32. Each element of $X_1$ gives rise to $2(2n-1)$ elements of $X_2$. By Exercise 27, $|X_1| = C_{n-1}$. By Exercise 31, all elements of $X_2$ are included. Therefore $|X_2| = C_{n-1}[2(2n-1)]$. By Exercise 28, $|X_2| = (n+1)C_n$ and the conclusion follows.

34. The hint provides a one-to-one correspondence between $n$-edge ordered trees and strings of $n$ zeros and $n$ ones in which, reading from the left, the number of ones is always greater than or equal to the number of zeros. The number of such strings is $C_n$, since these strings also encode paths in an $n \times n$ grid from the lower-left corner to the upper-right corner that never go above the diagonal from the lower-left corner to the upper-right corner (see Example 6.2.23). The encoding is obtained by interpreting a one as a move right and a zero as a move up.

## Section 9.9

2. The second player always wins. If the first player leaves $\{1, 3\}$ or $\{0, 3\}$, leave one. If the first player leaves $\{2, 3\}$, leave $\{2, 2\}$. After the first player moves, the second player can leave one. Part of the tree is



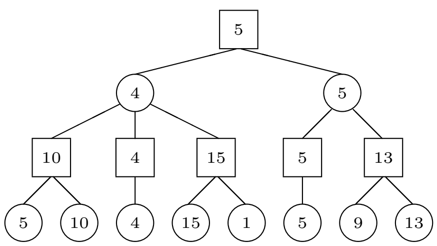
3. The tree is the same as Figure 9.9.1. The terminal vertices are assigned values as in Figure 9.9.2 with 0 and 1 interchanged. After applying the minimax procedure, the root receives the value 1; thus the first player will always win. The optimal strategy is to first leave $\{2, 2\}$. If the second player leaves only one pile, take it; otherwise, leave $\{1, 1\}$.

5. The tree is the same as in Exercise 1. The terminal vertices are assigned values as in the hint for Exercise 1 with 0 and 1 interchanged. After applying the minimax procedure, the root receives the value 1; thus the first player will always win. The optimal strategy is take 2. No matter how many player 2 chooses, player 1 can take the rest.

6. Figure 9.9.2

8. The strategy for winning play is: Play nim′ exactly like nim unless the move would leave an odd number of singleton piles and no other pile. In this case, leave an even number of piles.

10.




11. The value of the root is 10.

13. The value of the root is 9.

16. $4 - 2 = 2$ 17. $1 - 1 = 0$

20. No. Assign a larger value to a winning position.

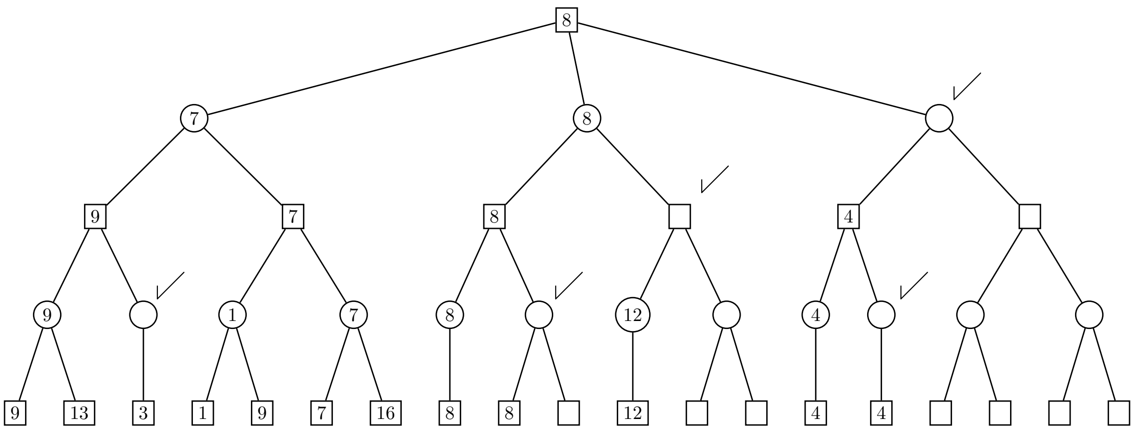
21. Input: the root $pt$ of a game tree, the level $pt\_level$ of $pt$, the maximum level $n$ to which the search is to be conducted, and an evaluation function $E$

Output: the game tree with the values of the vertices stored in the vertices

```
minimax(pt, pt_level, n, E) {
   if (pt_level == n) {
      contents(pt) = E(pt)
      return
   }
   let c_1, ..., c_k be the children of pt
   for i = 1 to k {
      minimax(c_i, pt_level + 1, n, E)
      e_i = contents(c_i)
   }
   if (pt is a box vertex)
      contents(pt) = max{e_1, ..., e_k}
```

else
  $contents(pt) = \min\{e_1, \ldots, e_k\}$
}

24. We first obtain the values 9, 6, 7 for the children of the root. Thus we order the children of the root 9, 7, 6 and use alpha-beta pruning to obtain




29–30. It is possible to always force a draw in Mu Torere, see P. D. Straffin, Jr., "Position graphs for Pong Hau K'i and Mu Torere," *Math. Mag.*, 68 (1995), 382–386, and "Corrected figure for position graphs for Pong Hau K'i and Mu Torere," *Math. Mag.*, 69 (1996), 65.