19. $f(n) = (4n^2 - 1)/3$ 21. $O(n^4)$ 23. $O(n)$ 25. Using just two comparisons, the algorithm is able to narrow the search for $m$ down to the first half or the second half of the original sequence. Since the length of the sequence is cut in half each time, only about $2 \log_2 n$ comparisons are needed in all. 27. **a)** $18n + 18$ **b)** 18 **c)** 0 29. $\Delta(a_n b_n) = a_{n+1}b_{n+1} - a_n b_n = a_{n+1}(b_{n+1} - b_n) + b_n(a_{n+1} - a_n) = a_{n+1}\Delta b_n + b_n \Delta a_n$ 31. **a)** Let $G(x) = \sum_{n=0}^{\infty} a_n x^n$. Then $G'(x) = \sum_{n=1}^{\infty} n a_n x^{n-1} = \sum_{n=0}^{\infty}(n+1)a_{n+1}x^n$. Therefore, $G'(x) - G(x) = \sum_{n=0}^{\infty}[(n+1)a_{n+1} - a_n]x^n = \sum_{n=0} x^n/n! = e^x$, as desired. That $G(0) = a_0 = 1$ is given. **b)** We have $[e^{-x}G(x)]' = e^{-x}G'(x) - e^{-x}G(x) = e^{-x}[G'(x) - G(x)] = e^{-x} \cdot e^x = 1$. Hence, $e^{-x}G(x) = x + c$, where $c$ is a constant. Consequently, $G(x) = xe^x + ce^x$. Because $G(0) = 1$, it follows that $c = 1$. **c)** We have $G(x) = \sum_{n=0}^{\infty} x^{n+1}/n! + \sum_{n=0}^{\infty} x^n/n! = \sum_{n=1}^{\infty} x^n/(n-1)! + \sum_{n=0}^{\infty} x^n/n!$. Therefore, $a_n = 1/(n-1)! + 1/n!$ for all $n \geq 1$, and $a_0 = 1$. 33. 7 35. 110 37. 0 39. **a)** 19 **b)** 65 **c)** 122 **d)** 167 **e)** 168 41. $D_{n-1}/(n-1)!$ 43. 11/32

# CHAPTER 9

## Section 9.1

1. **a)** $\{(0, 0), (1, 1), (2, 2), (3, 3)\}$ **b)** $\{(1, 3), (2, 2), (3, 1), (4, 0)\}$ **c)** $\{(1, 0), (2, 0), (2, 1), (3, 0), (3, 1), (3, 2), (4, 0), (4, 1), (4, 2), (4, 3)\}$ **d)** $\{(1, 0), (1, 1), (1, 2), (1, 3), (2, 0), (2, 2), (3, 0), (3, 3), (4, 0)\}$ **e)** $\{(0, 1), (1, 0), (1, 1), (1, 2), (1, 3), (2, 1), (2, 3), (3, 1), (3, 2), (4, 1), (4, 3)\}$ **f)** $\{(1, 2), (2, 1), (2, 2)\}$ 3. **a)** Transitive **b)** Reflexive, symmetric, transitive **c)** Symmetric **d)** Antisymmetric **e)** Reflexive, symmetric, antisymmetric, transitive **f)** None of these properties 5. **a)** Reflexive, transitive **b)** Symmetric **c)** Symmetric **d)** Symmetric 7. **a)** Symmetric **b)** Symmetric, transitive **c)** Symmetric **d)** Reflexive, symmetric, transitive **e)** Reflexive, transitive **f)** Reflexive, symmetric, transitive **g)** Antisymmetric **h)** Antisymmetric, transitive 9. Each of the three properties is vacuously satisfied. 11. (c), (d), (f) 13. **a)** Not irreflexive **b)** Not irreflexive **c)** Not irreflexive **d)** Not irreflexive 15. Yes, for instance $\{(1, 1)\}$ on $\{1, 2\}$ 17. $(a, b) \in R$ if and only if $a$ is taller than $b$ 19. (a) 21. None 23. $\forall a \forall b\, [(a, b) \in R \rightarrow (b, a) \notin R]$ 25. $2^{mn}$ 27. **a)** $\{(a, b) \mid b$ divides $a\}$ **b)** $\{(a, b) \mid a$ does not divide $b\}$ 29. The graph of $f^{-1}$ 31. **a)** $\{(a, b) \mid a$ is required to read or has read $b\}$ **b)** $\{(a, b) \mid a$ is required to read and has read $b\}$ **c)** $\{(a, b) \mid$ either $a$ is required to read $b$ but has not read it or $a$ has read $b$ but is not required to$\}$ **d)** $\{(a, b) \mid a$ is required to read $b$ but has not read it$\}$ **e)** $\{(a, b) \mid a$ has read $b$ but is not required to$\}$ 33. $S \circ R = \{(a, b) \mid a$ is a parent of $b$ and $b$ has a sibling$\}$, $R \circ S = \{(a, b) \mid a$ is an aunt or uncle of $b\}$ 35. **a)** $\mathbf{R}^2$ **b)** $R_6$ **c)** $R_3$ **d)** $R_3$ **e)** $\emptyset$ **f)** $R_1$ **g)** $R_4$ **h)** $R_4$ 37. **a)** $R_1$ **b)** $R_2$ **c)** $R_3$ **d)** $\mathbf{R}^2$ **e)** $R_3$ **f)** $\mathbf{R}^2$ **g)** $\mathbf{R}^2$ **h)** $\mathbf{R}^2$ 39. $b$ got his or her doctorate under someone who got his or her doctorate under $a$; there is a sequence of $n + 1$ people, starting with $a$ and ending with $b$, such that each is the advisor of the next person in the sequence 41. **a)** $\{(a, b) \mid a - b \equiv 0, 3, 4, 6, , 8,$ or $9 \pmod{12}\}$ **b)** $\{(a, b) \mid a \equiv b \pmod{12}\}$ **c)** $\{(a, b) \mid a - b \equiv 3, 6,$ or $9 \pmod{12}\}$ **d)** $\{(a, b) \mid a - b \equiv 4$ or $8 \pmod{12}\}$ **e)** $\{(a, b) \mid a - b \equiv 3, 4, 6, 8,$ or $9 \pmod{12}\}$ 43. 8 45. **a)** 65,536 **b)** 32,768 47. **a)** $2^{n(n+1)/2}$ **b)** $2^n 3^{n(n-1)/2}$ **c)** $3^{n(n-1)/2}$ **d)** $2^{n(n-1)}$ **e)** $2^{n(n-1)/2}$ **f)** $2^{n^2} - 2 \cdot 2^{n(n-1)}$ 49. There may be no such $b$. 51. If $R$ is symmetric and $(a, b) \in R$, then $(b, a) \in R$, so $(a, b) \in R^{-1}$. Hence, $R \subseteq R^{-1}$. Similarly, $R^{-1} \subseteq R$. So $R = R^{-1}$. Conversely, if $R = R^{-1}$ and $(a, b) \in R$, then $(a, b) \in R^{-1}$, so $(b, a) \in R$. Thus $R$ is symmetric. 53. $R$ is reflexive if and only if $(a, a) \in R$ for all $a \in A$ if and only if $(a, a) \in R^{-1}$ [because $(a, a) \in R$ if and only if $(a, a) \in R^{-1}$] if and only if $R^{-1}$ is reflexive. 55. Use mathematical induction. The result is trivial for $n = 1$. Assume $R^n$ is reflexive and transitive. By Theorem 1, $R^{n+1} \subseteq R$. To see that $R \subseteq R^{n+1} = R^n \circ R$, let $(a, b) \in R$. By the inductive hypothesis, $R^n = R$ and hence, is reflexive. Thus $(b, b) \in R^n$. Therefore $(a, b) \in R^{n+1}$. 57. Use mathematical induction. The result is trivial for $n = 1$. Assume $R^n$ is reflexive. Then $(a, a) \in R^n$ for all $a \in A$ and $(a, a) \in R$. Thus $(a, a) \in R^n \circ R = R^{n+1}$ for all $a \in A$. 59. No, for instance, take $R = \{(1, 2), (2, 1)\}$.

## Section 9.2

1. $\{(1, 2, 3), (1, 2, 4), (1, 3, 4), (2, 3, 4)\}$ 3. (Nadir, 122, 34, Detroit, 08:10), (Acme, 221, 22, Denver, 08:17), (Acme, 122, 33, Anchorage, 08:22), (Acme, 323, 34, Honolulu 08:30), (Nadir, 199, 13, Detroit, 08:47), (Acme, 222, 22, Denver, 09:10), (Nadir, 322, 34, Detroit, 09:44) 5. Airline and flight number, airline and departure time 7. **a)** Yes **b)** No **c)** No 9. **a)** Social Security number **b)** There are no two people with the same name who happen to have the same street address. **c)** There are no two people with the same name living together. 11. (Nadir, 122, 34, Detroit, 08 : 10), (Nadir, 199, 13, Detroit, 08 : 47), (Nadir, 322, 34, Detroit, 09 : 44) 13. (Nadir, 122, 34, Detroit, 08 : 10), (Nadir, 199, 13, Detroit, 08 : 47), (Nadir, 322, 34, Detroit, 09 : 44), (Acme, 221, 22, Denver, 08 : 17), (Acme, 222, 22, Denver, 09 : 10) 15. $P_{3,5,6}$

17.

| ***Airline*** | ***Destination*** |
|---|---|
| Nadir | Detroit |
| Acme | Denver |
| Acme | Anchorage |
| Acme | Honolulu |

19.

| *Supplier* | *Part_ number* | *Project* | *Quantity* | *Color_ code* |
|---|---|---|---|---|
| 23 | 1092 | 1 | 2 | 2 |
| 23 | 1101 | 3 | 1 | 1 |
| 23 | 9048 | 4 | 12 | 2 |
| 31 | 4975 | 3 | 6 | 2 |
| 31 | 3477 | 2 | 25 | 2 |
| 32 | 6984 | 4 | 10 | 1 |
| 32 | 9191 | 2 | 80 | 4 |
| 33 | 1001 | 1 | 14 | 8 |

21. Both sides of this equation pick out the subset of $R$ consisting of those $n$-tuples satisfying both conditions $C_1$ and $C_2$. 23. Both sides of this equation pick out the set of $n$-tuples that are in $R$, are in $S$, and satisfy condition $C$. 25. Both sides of this equation pick out the $m$-tuples consisting of $i_1$th, $i_2$th, ..., $i_m$th components of $n$-tuples in either $R$ or $S$. 27. Let $R = \{(a, b)\}$ and $S = \{(a, c)\}$, $n = 2$, $m = 1$, and $i_1 = 1$; $P_1(R - S) = \{(a)\}$, but $P_1(R) - P_1(S) = \emptyset$. 29. **a)** $J_2$ followed by $P_{1,3}$ **b)** $(23, 1)$, $(23, 3)$, $(31, 3)$, $(32, 4)$ 31. There is no primary key.

## Section 9.3

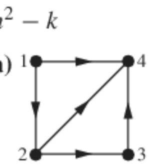
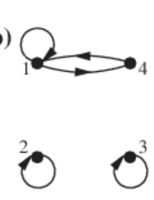
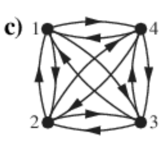
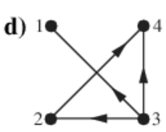
1. **a)** $\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ **b)** $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ **c)** $\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ **d)** $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$

3. **a)** $(1, 1)$, $(1, 3)$, $(2, 2)$, $(3, 1)$, $(3, 3)$ **b)** $(1, 2)$, $(2, 2)$, $(3, 2)$ **c)** $(1, 1)$, $(1, 2)$, $(1, 3)$, $(2, 1)$, $(2, 3)$, $(3, 1)$, $(3, 2)$, $(3, 3)$ 5. The relation is irreflexive if and only if the main diagonal of the matrix contains only 0s. 7. **a)** Reflexive, symmetric, transitive **b)** Antisymmetric, transitive **c)** Symmetric 9. **a)** 4950 **b)** 9900 **c)** 99 **d)** 100 **e)** 1 11. Change each 0 to a 1 and each 1 to a 0.

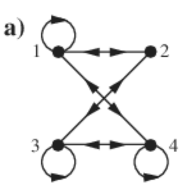
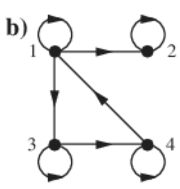
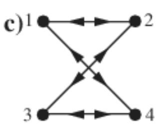
13. **a)** $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$ **b)** $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ **c)** $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

15. **a)** $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$ **b)** $\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ **c)** $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

17. $n^2 - k$

19. **a)**  **b)**  **c)**  **d)** 

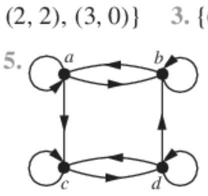
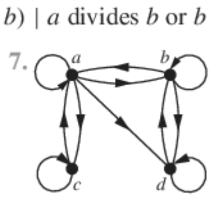
21. For simplicity we have indicated pairs of edges between the same two vertices in opposite directions by using a double arrowhead, rather than drawing two separate lines.

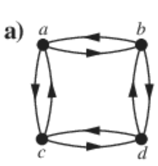
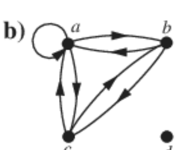
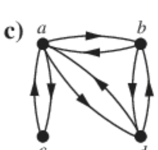
**a)**  **b)**  **c)** 

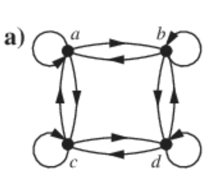
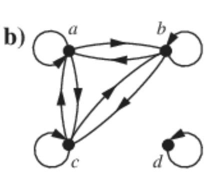
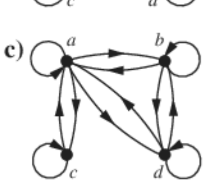
23. $\{(a, b), (a, c), (b, c), (c, b)\}$ 25. $(a, c)$, $(b, a)$, $(c, d)$, $(d, b)$ 27. $\{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (d, d)\}$ 29. The relation is asymmetric if and only if the directed graph has no loops and no closed paths of length 2. 31. Exercise 23: irreflexive. Exercise 24: reflexive, antisymmetric, transitive. Exercise 25: irreflexive, antisymmetric. 33. Reverse the direction on every edge in the digraph for $R$. 35. Proof by mathematical induction. *Basis step:* Trivial for $n = 1$. *Inductive step:* Assume true for $k$. Because $R^{k+1} = R^k \circ R$, its matrix is $\mathbf{M}_R \odot \mathbf{M}_{R^k}$. By the inductive hypothesis this is $\mathbf{M}_R \odot \mathbf{M}_R^{[k]} = \mathbf{M}_R^{[k+1]}$.

## Section 9.4

1. **a)** $\{(0, 0), (0, 1), (1, 1), (1, 2), (2, 0), (2, 2), (3, 0), (3, 3)\}$ **b)** $\{(0, 1), (0, 2), (0, 3), (1, 0), (1, 1), (1, 2), (2, 0), (2, 1), (2, 2), (3, 0)\}$ 3. $\{(a, b) \mid a \text{ divides } b \text{ or } b \text{ divides } a\}$

5.  7. 

9. **a)**  **b)**  **c)** 

11. **a)**  **b)**  **c)** 

13. The symmetric closure of $R$ is $R \cup R^{-1}$. $\mathbf{M}_{R \cup R^{-1}} = \mathbf{M}_R \vee \mathbf{M}_{R^{-1}} = \mathbf{M}_R \vee \mathbf{M}_R^t$. 15. Only when $R$ is irreflexive,

in which case it is its own closure. 17. $a, a, a, a; a, b, e, a; a, d, e, a; b, c, c, b; b, e, a, b; c, b, c, c; c, c, b, c; c, c, c, c; d, e, a, d; d, e, e, d; e, a, b, e; e, a, d, e; e, d, e, e; e, e, d, e; e, e, e, e$ 19. **a)** $\{(1, 1), (1, 5), (2, 3), (3, 1), (3, 2), (3, 3), (3, 4), (4, 1), (4, 5), (5, 3), (5, 4)\}$ **b)** $\{(1, 1), (1, 2), (1, 3), (1, 4), (2, 1), (2, 5), (3, 1), (3, 3), (3, 4), (3, 5), (4, 1), (4, 2), (4, 3), (4, 4), (5, 1), (5, 3), (5, 5)\}$ **c)** $\{(1, 1), (1, 3), (1, 4), (1, 5), (2, 1), (2, 2), (2, 3), (2, 4), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (4, 1), (4, 3), (4, 4), (4, 5), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)\}$ **d)** $\{(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (2, 1), (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (4, 1), (4, 2), (4, 3), (4, 4), (4, 5), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)\}$ **e)** $\{(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (4, 1), (4, 2), (4, 3), (4, 4), (4, 5), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)\}$ **f)** $\{(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (2, 1), (2, 2), (2, 3), (2, 4), (2, 5), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (4, 1), (4, 2), (4, 3), (4, 4), (4, 5), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)\}$ 21. **a)** If there is a student $c$ who shares a class with $a$ and a class with $b$ **b)** If there are two students $c$ and $d$ such that $a$ and $c$ share a class, $c$ and $d$ share a class, and $d$ and $b$ share a class **c)** If there is a sequence $s_0, \ldots, s_n$ of students with $n \geq 1$ such that $s_0 = a$, $s_n = b$, and for each $i = 1, 2, \ldots, n$, $s_i$ and $s_{i-1}$ share a class 23. The result follows from $(R^*)^{-1} = \left(\bigcup_{n=1}^{\infty} R^n\right)^{-1} = \bigcup_{n=1}^{\infty}(R^n)^{-1} = \bigcup_{n=1}^{\infty} R^n = R^*$.

25. **a)** $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ **b)** $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$

**c)** $\begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ **d)** $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

27. Answers same as for Exercise 25. 29. **a)** $\{(1, 1), (1, 2), (1, 4), (2, 2), (3, 3), (4, 1), (4, 2), (4, 4)\}$ **b)** $\{(1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (2, 4), (3, 3), (4, 1), (4, 2), (4, 4)\}$ **c)** $\{(1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (2, 4), (3, 3), (4, 1), (4, 2), (4, 4)\}$ 31. Algorithm 1: $O(n^{3.8})$; Algorithm 2: $O(n^3)$ 33. Initialize with $\mathbf{A} := \mathbf{M}_R \vee \mathbf{I}_n$ and loop only for $i := 2$ to $n - 1$. 35. **a)** Because $R$ is reflexive, every relation containing it must also be reflexive. **b)** Both $\{(0, 0), (0, 1), (0, 2), (1, 1), (2, 2)\}$ and $\{(0, 0), (0, 1), (1, 0), (1, 1), (2, 2)\}$ contain $R$ and have an odd number of elements, but neither is a subset of the other.

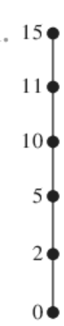
## Section 9.5

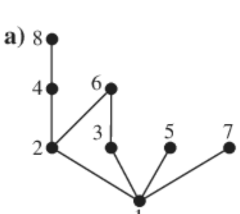
1. **a)** Equivalence relation **b)** Not reflexive, not transitive **c)** Equivalence relation **d)** Not transitive **e)** Not symme- tric, not transitive 3. **a)** Equivalence relation **b)** Not transitive **c)** Not reflexive, not symmetric, not transitive **d)** Equivalence relation **e)** Not reflexive, not transitive 5. Many answers are possible. (1) Two buildings are equivalent if they were opened during the same year; an equivalence class consists of the set of buildings opened in a given year (as long as there was at least one building opened that year). (2) Two buildings are equivalent if they have the same number of stories; the equivalence classes are the set of 1-story buildings, the set of 2-story buildings, and so on (one class for each $n$ for which there is at least one $n$-story building). (3) Every building in which you have a class is equivalent to every building in which you have a class (including itself), and every building in which you don't have a class is equivalent to every building in which you don't have a class (including itself); there are two equivalence classes—the set of buildings in which you have a class and the set of buildings in which you don't have a class (assuming these are nonempty). 7. The statement "$p$ is equivalent to $q$" means that $p$ and $q$ have the same entries in their truth tables. $R$ is reflexive, because $p$ has the same truth table as $p$. $R$ is symmetric, for if $p$ and $q$ have the same truth table, then $q$ and $p$ have the same truth table. If $p$ and $q$ have the same entries in their truth tables and $q$ and $r$ have the same entries in their truth tables, then $p$ and $r$ also do, so $R$ is transitive. The equivalence class of $\mathbf{T}$ is the set of all tautologies; the equivalence class of $\mathbf{F}$ is the set of all contradictions. 9. **a)** $(x, x) \in R$ because $f(x) = f(x)$. Hence, $R$ is reflexive. $(x, y) \in R$ if and only if $f(x) = f(y)$, which holds if and only if $f(y) = f(x)$ if and only if $(y, x) \in R$. Hence, $R$ is symmetric. If $(x, y) \in R$ and $(y, z) \in R$, then $f(x) = f(y)$ and $f(y) = f(z)$. Hence, $f(x) = f(z)$. Thus, $(x, z) \in R$. It follows that $R$ is transitive. **b)** The sets $f^{-1}(b)$ for $b$ in the range of $f$ 11. Let $x$ be a bit string of length 3 or more. Because $x$ agrees with itself in the first three bits, $(x, x) \in R$. Hence, $R$ is reflexive. Suppose that $(x, y) \in R$. Then $x$ and $y$ agree in the first three bits. Hence, $y$ and $x$ agree in the first three bits. Thus, $(y, x) \in R$. If $(x, y)$ and $(y, z)$ are in $R$, then $x$ and $y$ agree in the first three bits, as do $y$ and $z$. Hence, $x$ and $z$ agree in the first three bits. Hence, $(x, z) \in R$. It follows that $R$ is transitive. 13. This follows from Exercise 9, where $f$ is the function that takes a bit string of length 3 or more to the ordered pair with its first bit as the first component and the third bit as its second component. 15. For reflexivity, $((a, b), (a, b)) \in R$ because $a + b = b + a$. For symmetry, if $((a, b), (c, d)) \in R$, then $a + d = b + c$, so $c + b = d + a$, so $((c, d), (a, b)) \in R$. For transitivity, if $((a, b), (c, d)) \in R$ and $((c, d), (e, f)) \in R$, then $a+d = b+c$ and $c+e = d+f$, so $a + d + c + e = b + c + d + f$, so $a + e = b + f$, so $((a, b), (e, f)) \in R$. An easier solution is to note that by algebra, the given condition is the same as the condition that $f((a, b)) = f((c, d))$, where $f((x, y)) = x - y$; therefore by Exercise 9 this is an equivalence relation. 17. **a)** This follows from Exercise 9, where the function $f$ from the set of differentiable functions (from $\mathbf{R}$ to $\mathbf{R}$) to the set of functions (from $\mathbf{R}$ to $\mathbf{R}$) is the differentiation operator. **b)** The set of all functions of the form $g(x) = x^2 + C$ for some constant $C$ 19. This follows from Exercise 9, where the function $f$ from the set of all URLs to the set of all Web pages is the function that assigns to each URL the Web page for that URL. 21. No 23. No 25. $R$ is reflexive because a bit string $s$

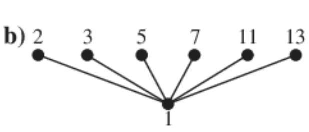
has the same number of 1s as itself. $R$ is symmetric because $s$ and $t$ having the same number of 1s implies that $t$ and $s$ do. $R$ is transitive because $s$ and $t$ having the same number of 1s, and $t$ and $u$ having the same number of 1s implies that $s$ and $u$ have the same number of 1s. **27. a)** The sets of people of the same age **b)** The sets of people with the same two parents **29.** The set of all bit strings with exactly two 1s. **31. a)** The set of all bit strings of length 3 **b)** The set of all bit strings of length 4 that end with a 1 **c)** The set of all bit strings of length 5 that end 11 **d)** The set of all bit strings of length 8 that end 10101 **33.** Each of the 15 bit strings of length less than four is in an equivalence class by itself: $[\lambda]_{R_4} = \{\lambda\}$, $[0]_{R_4} = \{0\}$, $[1]_{R_4} = \{1\}$, $[00]_{R_4} = \{00\}$, $[01]_{R_4} = \{01\}$, ..., $[111]_{R_4} = \{111\}$. The remaining 16 equivalence classes are determined by the bit strings of length 4: $[0000]_{R_4} = \{0000, 00000, 00001, 000000, 000001, 000010, 000011, 0000000, \ldots\}$, $[0001]_{R_4} = \{0001, 00010, 00011, 000100, 000101, 000110, 000111, 0001000, \ldots\}$, ..., $[1111]_{R_4} = \{1111, 11110, 11111, 111100, 111101, 111110, 111111, 1111000, \ldots\}$ **35. a)** $[2]_5 = \{i \mid i \equiv 2 \pmod 5\} = \{\ldots, -8, -3, 2, 7, 12, \ldots\}$ **b)** $[3]_5 = \{i \mid i \equiv 3 \pmod 5\} = \{\ldots, -7, -2, 3, 8, 13, \ldots\}$ **c)** $[6]_5 = \{i \mid i \equiv 6 \pmod 5\} = \{\ldots, -9, -4, 1, 6, 11, \ldots\}$ **d)** $[-3]_5 = \{i \mid i \equiv -3 \pmod 5\} = \{\ldots, -8, -3, 2, 7, 12, \ldots\}$ **37.** $\{6n+k \mid n \in \mathbf{Z}\}$ for $k \in \{0, 1, 2, 3, 4, 5\}$ **39. a)** $[(1, 2)] = \{(a, b) \mid a - b = -1\} = \{(1, 2), (3, 4), (4, 5), (5, 6), \ldots\}$ **b)** Each equivalence class can be interpreted as an integer (negative, positive, or zero); specifically, $[(a, b)]$ can be interpreted as $a - b$. **41. a)** No **b)** Yes **c)** Yes **d)** No **43.** (a), (c), (e) **45.** (b), (d), (e) **47. a)** $\{(0, 0), (1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (3, 4), (3, 5), (4, 3), (4, 4), (4, 5), (5, 3), (5, 4), (5, 5)\}$ **b)** $\{(0, 0), (0, 1), (1, 0), (1, 1), (2, 2), (2, 3), (3, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5)\}$ **c)** $\{(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2), (2, 0), (2, 1), (2, 2), (3, 3), (3, 4), (3, 5), (4, 3), (4, 4), (4, 5), (5, 3), (5, 4), (5, 5)\}$ **d)** $\{(0, 0), (1, 1), (2, 2), (3, 3), (4, 4), (5, 5)\}$ **49.** $[0]_6 \subseteq [0]_3$, $[1]_6 \subseteq [1]_3$, $[2]_6 \subseteq [2]_3$, $[3]_6 \subseteq [0]_3$, $[4]_6 \subseteq [1]_3$, $[5]_6 \subseteq [2]_3$ **51.** Let $A$ be a set in the first partition. Pick a particular element $x$ of $A$. The set of all bit strings of length 16 that agree with $x$ on the last four bits is one of the sets in the second partition, and clearly every string in $A$ is in that set. **53.** We claim that each equivalence class $[x]_{R_{31}}$ is a subset of the equivalence class $[x]_{R_8}$. To show this, choose an arbitrary element $y \in [x]_{R_{31}}$. Then $y$ is equivalent to $x$ under $R_{31}$, so either $y = x$ or $y$ and $x$ are each at least 31 characters long and agree on their first 31 characters. Because strings that are at least 31 characters long and agree on their first 31 characters perforce are at least 8 characters long and agree on their first 8 characters, we know that either $y = x$ or $y$ and $x$ are each at least 8 characters long and agree on their first 8 characters. This means that $y$ is equivalent to $x$ under $R_8$, so $y \in [x]_{R_8}$. **55.** $\{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c), (d, d), (d, e), (e, d), (e, e)\}$ **57. a)** $\mathbf{Z}$ **b)** $\{n + \frac{1}{2} \mid n \in \mathbf{Z}\}$ **59. a)** $R$ is reflexive because any coloring can be obtained from itself via a 360-degree rotation. To see that $R$ is symmetric and transitive, use the fact that each rotation is the composition of two reflections and conversely the composition of two reflections is a rotation. Hence, $(C_1, C_2)$ belongs to $R$ if and only if $C_2$ can be obtained from $C_1$ by a composition of reflections. So if $(C_1, C_2)$ belongs to $R$, so does $(C_2, C_1)$ because the inverse of the composition of reflections is also a composition of reflections (in the opposite order). Hence, $R$ is symmetric. To see that $R$ is transitive, suppose $(C_1, C_2)$ and $(C_2, C_3)$ belong to $R$. Taking the composition of the reflections in each case yields a composition of reflections, showing that $(C_1, C_3)$ belongs to $R$. **b)** We express colorings with sequences of length four, with $r$ and $b$ denoting red and blue, respectively. We list letters denoting the colors of the upper left square, upper right square, lower left square, and lower right square, in that order. The equivalence classes are: $\{rrrr\}$, $\{bbbb\}$, $\{rrrb, rrbr, rbrr, brrr\}$, $\{bbbr, bbrb, brbb, rbbb\}$, $\{rbbr, brrb\}$, $\{rrbb, brbr, bbrr, rbrb\}$. **61.** 5 **63.** Yes **65.** $R$ **67.** First form the reflexive closure of $R$, then form the symmetric closure of the reflexive closure, and finally form the transitive closure of the symmetric closure of the reflexive closure. **69.** $p(0) = 1$, $p(1) = 1$, $p(2) = 2$, $p(3) = 5$, $p(4) = 15$, $p(5) = 52$, $p(6) = 203$, $p(7) = 877$, $p(8) = 4140$, $p(9) = 21147$, $p(10) = 115975$

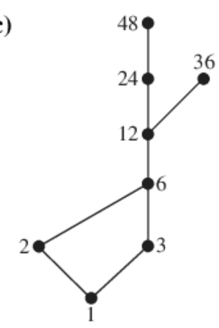
## Section 9.6

**1. a)** Is a partial ordering **b)** Not antisymmetric, not transitive **c)** Is a partial ordering **d)** Is a partial ordering **e)** Not antisymmetric, not transitive **3. a)** No **b)** No **c)** Yes **5. a)** Yes **b)** No **c)** Yes **d)** No **7. a)** No **b)** Yes **c)** No **9.** No **11.** Yes **13. a)** $\{(0, 0), (1, 0), (1, 1), (2, 0), (2, 1), (2, 2)\}$ **b)** $(\mathbf{Z}, \leq)$ **c)** $(P(\mathbf{Z}), \subseteq)$ **d)** $(\mathbf{Z}^+$, "is a multiple of") **15. a)** $\{0\}$ and $\{1\}$, for instance **b)** 4 and 6, for instance **17. a)** $(1, 1, 2) < (1, 2, 1)$ **b)** $(0, 1, 2, 3) < (0, 1, 3, 2)$ **c)** $(0, 1, 1, 1, 0) < (1, 0, 1, 0, 1)$ **19.** $0 < 0001 < 001 < 01 < 010 < 0101 < 011 < 11$

**21.**



**23. a)**



**b)**


c) 


d) 


25. $(a, b)$, $(a, c)$, $(a, d)$, $(b, c)$, $(b, d)$, $(a, a)$, $(b, b)$, $(c, c)$, $(d, d)$ 27. $(a, a)$, $(a, g)$, $(a, d)$, $(a, e)$, $(a, f)$, $(b, b)$, $(b, g)$, $(b, d)$, $(b, e)$, $(b, f)$, $(c, c)$, $(c, g)$, $(c, d)$, $(c, e)$, $(c, f)$, $(g, d)$, $(g, e)$, $(g, f)$, $(g, g)$, $(d, d)$, $(e, e)$, $(f, f)$ 29. $(\emptyset, \{a\})$, $(\emptyset, \{b\})$, $(\emptyset, \{c\})$, $(\{a\}, \{a, b\})$, $(\{a\}, \{a, c\})$, $(\{b\}, \{a, b\})$, $(\{b\}, \{b, c\})$, $(\{c\}, \{a, c\})$, $(\{c\}, \{b, c\})$, $(\{a, b\}, \{a, b, c\})$, $(\{a, c\}, \{a, b, c\})(\{b, c\}, \{a, b, c\})$ 31. Let $(S, \preccurlyeq)$ be a finite poset. We will show that this poset is the reflexive transitive closure of its covering relation. Suppose that $(a, b)$ is in the reflexive transitive closure of the covering relation. Then $a = b$ or $a \prec b$, so $a \preccurlyeq b$, or else there is a sequence $a_1, a_2, \ldots, a_n$ such that $a \prec a_1 \prec a_2 \prec \cdots \prec a_n \prec b$, in which case again $a \preccurlyeq b$ by the transitivity of $\preccurlyeq$. Conversely, suppose that $a \prec b$. If $a = b$ then $(a, b)$ is in the reflexive transitive closure of the covering relation. If $a \prec b$ and there is no $z$ such that $a \prec z \prec b$, then $(a, b)$ is in the covering relation and therefore in its reflexive transitive closure. Otherwise, let $a \prec a_1 \prec a_2 \prec \cdots \prec a_n \prec b$ be a longest possible sequence of this form (which exists because the poset is finite). Then no intermediate elements can be inserted, so each pair $(a, a_1)$, $(a_1, a_2)$, $\ldots$, $(a_n, b)$ is in the covering relation, so again $(a, b)$ is in its reflexive transitive closure. 33. **a)** 24, 45 **b)** 3, 5 **c)** No **d)** No **e)** 15, 45 **f)** 15 **g)** 15, 5, 3 **h)** 15 35. **a)** $\{1, 2\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$ **b)** $\{1\}$, $\{2\}$, $\{4\}$ **c)** No **d)** No **e)** $\{2, 4\}$, $\{2, 3, 4\}$ **f)** $\{2, 4\}$ **g)** $\{3, 4\}$, $\{4\}$ **h)** $\{3, 4\}$ 37. Because $(a, b) \preccurlyeq (a, b)$, $\preccurlyeq$ is reflexive. If $(a_1, a_2) \preccurlyeq (b_1, b_2)$ and $(a_1, a_2) \neq (b_1, b_2)$, either $a_1 \prec b_1$, or $a_1 = b_1$ and $a_2 \prec b_2$. In either case, $(b_1, b_2)$ is not less than or equal to $(a_1, a_2)$. Hence, $\preccurlyeq$ is antisymmetric. Suppose that $(a_1, a_2) \prec (b_1, b_2) \prec (c_1, c_2)$. Then if $a_1 \prec b_1$ or $b_1 \prec c_1$, we have $a_1 \prec c_1$, so $(a_1, a_2) \prec (c_1, c_2)$, but if $a_1 = b_1 = c_1$, then $a_2 \prec b_2 \prec c_2$, which implies that $(a_1, a_2) \prec (c_1, c_2)$. Hence, $\preccurlyeq$ is transitive. 39. Because $(s, t) \preccurlyeq (s, t)$, $\preccurlyeq$ is reflexive. If $(s, t) \preccurlyeq (u, v)$ and $(u, v) \preccurlyeq (s, t)$, then $s \preccurlyeq u \preccurlyeq s$ and $t \preccurlyeq v \preccurlyeq t$; hence, $s = u$ and $t = v$. Hence, $\preccurlyeq$ is antisymmetric. Suppose that $(s, t) \preccurlyeq (u, v) \preccurlyeq (w, x)$. Then $s \preccurlyeq u$, $t \preccurlyeq v$, $u \preccurlyeq w$, and $v \preccurlyeq x$. It follows that $s \preccurlyeq w$ and $t \preccurlyeq x$. Hence, $(s, t) \preccurlyeq (w, x)$. Hence, $\preccurlyeq$ is transitive. 41. **a)** Suppose that $x$ is maximal and that $y$ is the largest element. Then $x \preccurlyeq y$. Because $x$ is not less than $y$, it follows that $x = y$. By Exercise 40(a) $y$ is unique. Hence, $x$ is unique. **b)** Suppose that $x$ is minimal and that $y$ is the smallest element. Then $x \succcurlyeq y$. Because $x$ is not greater than $y$, it follows that $x = y$. By Exercise 40(b) $y$ is unique. Hence, $x$ is unique. 43. **a)** Yes **b)** No **c)** Yes 45. Use mathematical induction. Let $P(n)$ be "Every subset with $n$ elements from a lattice has a least upper bound and a greatest lower bound." *Basis step:* $P(1)$ is true because the least upper bound and greatest lower bound of $\{x\}$ are both $x$. *Inductive step:* Assume that $P(k)$ is true. Let $S$ be a set with $k + 1$ elements. Let $x \in S$ and $S' = S - \{x\}$. Because $S'$ has $k$ elements, by the inductive hypothesis, it has a least upper bound $y$ and a greatest lower bound $a$. Now because we are in a lattice, there are elements $z = \text{lub}(x, y)$ and $b = \text{glb}(x, a)$. We are done if we can show that $z$ is the least upper bound of $S$ and $b$ is the greatest lower bound of $S$. To show that $z$ is the least upper bound of $S$, first note that if $w \in S$, then $w = x$ or $w \in S'$. If $w = x$ then $w \preccurlyeq z$ because $z$ is the least upper bound of $x$ and $y$. If $w \in S'$, then $w \preccurlyeq z$ because $w \preccurlyeq y$, which is true because $y$ is the least upper bound of $S'$, and $y \preccurlyeq z$, which is true because $z = \text{lub}(x, y)$. To see that $z$ is the least upper bound of $S$, suppose that $u$ is an upper bound of $S$. Note that such an element $u$ must be an upper bound of $x$ and $y$, but because $z = \text{lub}(x, y)$, it follows that $z \preccurlyeq u$. We omit the similar argument that $b$ is the greatest lower bound of $S$. 47. **a)** No **b)** Yes **c)** (*Proprietary*, {*Cheetah*, *Puma*}), (*Restricted*, {*Cheetah*, *Puma*}), (*Registered*, {*Cheetah*, *Puma*}), (*Proprietary*, {*Cheetah*, *Puma*, *Impala*}), (*Restricted*, {*Cheetah*, *Puma*, *Impala*}), (*Registered*, {*Cheetah*, *Puma*, *Impala*}) **d)** (*Non- proprietary*, {*Impala*, *Puma*}), (*Proprietary*, {*Impala*, *Puma*}), (*Restricted*, {*Impala*, *Puma*}), (*Nonproprietary*, {*Impala*}), (*Proprietary*, {*Impala*}), (*Restricted*, {*Impala*}), (*Nonproprietary*, {*Puma*}), (*Proprietary*, {*Puma*}), (*Restricted*, {*Puma*}), (*Nonproprietary*, $\emptyset$), (*Proprietary*, $\emptyset$), (*Restricted*, $\emptyset$) 49. Let $\Pi$ be the set of all partitions of a set $S$ with $P_1 \preccurlyeq P_2$ if $P_1$ is a refinement of $P_2$, that is, if every set in $P_1$ is a subset of a set in $P_2$. First, we show that $(\Pi, \preccurlyeq)$ is a poset. Because $P \preccurlyeq P$ for every partition $P$, $\preccurlyeq$ is reflexive. Now suppose that $P_1 \preccurlyeq P_2$ and $P_2 \preccurlyeq P_1$. Let $T \in P_1$. Because $P_1 \preccurlyeq P_2$, there is a set $T' \in P_2$ such that $T \subseteq T'$. Because $P_2 \preccurlyeq P_1$ there is a set $T'' \in P_1$ such that $T' \subseteq T''$. It follows that $T \subseteq T''$. But because $P_1$ is a partition, $T = T''$, which implies that $T = T'$ because $T \subseteq T' \subseteq T''$. Thus, $T \in P_2$. By reversing the roles of $P_1$ and $P_2$ it follows that every set in $P_2$ is also in $P_1$. Hence, $P_1 = P_2$ and $\preccurlyeq$ is antisymmetric. Next, suppose that $P_1 \preccurlyeq P_2$ and $P_2 \preccurlyeq P_3$. Let $T \in P_1$. Then there is a set $T' \in P_2$ such that $T \subseteq T'$. Because $P_2 \preccurlyeq P_3$ there is a set $T'' \in P_3$ such that $T' \subseteq T''$. This means that $T \subseteq T''$. Hence, $P_1 \preccurlyeq P_3$. It follows that $\preccurlyeq$ is transitive. The greatest lower bound of the partitions $P_1$ and $P_2$ is the partition $P$ whose subsets are the nonempty sets of the form $T_1 \cap T_2$ where $T_1 \in P_1$ and $T_2 \in P_2$. We omit the justification of this statement here. The least upper bound of the partitions $P_1$ and $P_2$ is the partition that corresponds to the equivalence relation in which $x \in S$ is related to $y \in S$ if there is a sequence $x = x_0, x_1, x_2, \ldots, x_n = y$ for some nonnegative integer $n$ such that for each $i$ from 1 to $n$, $x_{i-1}$ and $x_i$ are in the same element of $P_1$ or of $P_2$. We omit the details that this is an equivalence relation and the details of the proof that this is

the least upper bound of the two partitions. **51.** By Exercise 45 there is a least upper bound and a greatest lower bound for the entire finite lattice. By definition these elements are the greatest and least elements, respectively. **53.** The least element of a subset of $\mathbf{Z}^+ \times \mathbf{Z}^+$ is that pair that has the smallest possible first coordinate, and, if there is more than one such pair, that pair among those that has the smallest second coordinate. **55.** If $x$ is an integer in a decreasing sequence of elements of this poset, then at most $|x|$ elements can follow $x$ in the sequence, namely, integers whose absolute values are $|x| - 1, |x| - 2, \ldots, 1, 0$. Therefore there can be no infinite decreasing sequence. This is not a totally ordered set, because 5 and $-5$, for example, are incomparable. **57.** To find which of two rational numbers is larger, write them with a positive common denominator and compare numerators. To show that this set is dense, suppose that $x < y$ are two rational numbers. Then their average, i.e., $(x + y)/2$, is a rational number between them. **59.** Let $(S, \preccurlyeq)$ be a partially ordered set. It is enough to show that every nonempty subset of $S$ contains a least element if and only if there is no infinite decreasing sequence of elements $a_1, a_2, a_3, \ldots$ in $S$ (i.e., where $a_{i+1} \prec a_i$ for all $i$). An infinite decreasing sequence of elements clearly has no least element. Conversely, let $A$ be any nonempty subset of $S$ that has no least element. Because $A$ is nonempty, choose $a_1 \in A$. Because $a_1$ is not the least element of $A$, choose $a_2 \in A$ with $a_2 \prec a_1$. Because $a_2$ is not the least element of $A$, choose $a_3 \in A$ with $a_3 \prec a_2$. Continue in this manner, producing an infinite decreasing sequence in $S$. **61.** $a \prec_t b \prec_t c \prec_t d \prec_t e \prec_t f \prec_t g \prec_t h \prec_t i \prec_t j \prec_t k \prec_t l \prec_t m$ **63.** $1 \prec 5 \prec 2 \prec 4 \prec 12 \prec 20$, $1 \prec 2 \prec 5 \prec 4 \prec 12 \prec 20$, $1 \prec 2 \prec 4 \prec 5 \prec 12 \prec 20$, $1 \prec 2 \prec 4 \prec 12 \prec 5 \prec 20$, $1 \prec 5 \prec 2 \prec 4 \prec 20 \prec 12$, $1 \prec 2 \prec 5 \prec 4 \prec 20 \prec 12$, $1 \prec 2 \prec 4 \prec 5 \prec 20 \prec 12$ **65.** $A \prec C \prec E \prec B \prec D \prec F \prec G$, $A \prec E \prec C \prec B \prec D \prec F \prec G$, $C \prec A \prec E \prec B \prec D \prec F \prec G$, $C \prec E \prec A \prec B \prec D \prec F \prec G$, $E \prec A \prec C \prec B \prec D \prec F \prec G$, $E \prec C \prec A \prec B \prec D \prec F \prec G$, $A \prec C \prec B \prec E \prec D \prec F \prec G$, $C \prec A \prec B \prec E \prec D \prec F \prec G$, $A \prec C \prec B \prec D \prec E \prec F \prec G$, $C \prec A \prec B \prec D \prec E \prec F \prec G$, $A \prec C \prec E \prec B \prec F \prec D \prec G$, $A \prec E \prec C \prec B \prec F \prec D \prec G$, $C \prec A \prec E \prec B \prec F \prec D \prec G$, $C \prec E \prec A \prec B \prec F \prec D \prec G$, $E \prec A \prec C \prec B \prec F \prec D \prec G$, $E \prec C \prec A \prec B \prec F \prec D \prec G$, $A \prec C \prec B \prec E \prec F \prec D \prec G$, $C \prec A \prec B \prec E \prec F \prec D \prec G$ **67.** Determine user needs $\prec$ Write functional requirements $\prec$ Set up test sites $\prec$ Develop system requirements $\prec$ Write documentation $\prec$ Develop module $A$ $\prec$ Develop module $B$ $\prec$ Develop module $C$ $\prec$ Integrate modules $\prec$ $\alpha$ test $\prec$ $\beta$ test $\prec$ Completion

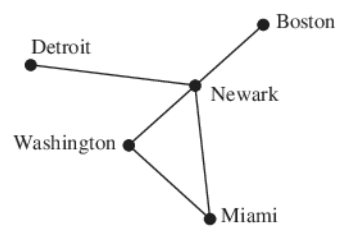
## Supplementary Exercises

**1. a)** Irreflexive (we do not include the empty string), symmetric **b)** Irreflexive, symmetric **c)** Irreflexive, antisymmetric, transitive **3.** $((a, b), (a, b)) \in R$ because $a + b = a + b$. Hence, $R$ is reflexive. If $((a, b), (c, d)) \in R$ then $a + d = b + c$, so that $c + b = d + a$. It follows that $((c, d), (a, b)) \in R$. Hence, $R$ is symmetric. Suppose that $((a, b), (c, d))$ and $((c, d), (e, f))$ belong to $R$. Then $a + d = b + c$ and $c + f = d + e$. Adding these two equations and subtracting $c + d$ from both sides gives $a + f = b + e$. Hence, $((a, b), (e, f))$ belongs to $R$. Hence, $R$ is transitive. **5.** Suppose that $(a, b) \in R$. Because $(b, b) \in R$ it follows that $(a, b) \in R^2$. **7.** Yes, yes **9.** Yes, yes **11.** Two records with identical keys in the projection would have identical keys in the original. **13.** $(\Delta \cup R)^{-1} = \Delta^{-1} \cup R^{-1} = \Delta \cup R^{-1}$ **15. a)** $R = \{(a, b), (a, c)\}$. The transitive closure of the symmetric closure of $R$ is $\{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c)\}$ and is different from the symmetric closure of the transitive closure of $R$, which is $\{(a, b), (a, c), (b, a), (c, a)\}$. **b)** Suppose that $(a, b)$ is in the symmetric closure of the transitive closure of $R$. We must show that $(a, b)$ is in the transitive closure of the symmetric closure of $R$. We know that at least one of $(a, b)$ and $(b, a)$ is in the transitive closure of $R$. Hence, there is either a path from $a$ to $b$ in $R$ or a path from $b$ to $a$ in $R$ (or both). In the former case, there is a path from $a$ to $b$ in the symmetric closure of $R$. In the latter case, we can form a path from $a$ to $b$ in the symmetric closure of $R$ by reversing the directions of all the edges in a path from $b$ to $a$, going backward. Hence, $(a, b)$ is in the transitive closure of the symmetric closure of $R$. **17.** The closure of $S$ with respect to property $\mathbf{P}$ is a relation with property $\mathbf{P}$ that contains $R$ because $R \subseteq S$. Hence, the closure of $S$ with respect to property $\mathbf{P}$ contains the closure of $R$ with respect to property $\mathbf{P}$. **19.** Use the basic idea of Warshall's algorithm, except let $w_{ij}^{[k]}$ equal the length of the longest path from $v_i$ to $v_j$ using interior vertices with subscripts not exceeding $k$, and equal to $-1$ if there is no such path. To find $w_{ij}^{[k]}$ from the entries of $\mathbf{W}_{k-1}$, determine for each pair $(i, j)$ whether there are paths from $v_i$ to $v_k$ and from $v_k$ to $v_j$ using no vertices labeled greater than $k$. If either $w_{ik}^{[k-1]}$ or $w_{kj}^{[k-1]}$ is $-1$, then such a pair of paths does not exist, so set $w_{ij}^{[k]} = w_{ij}^{[k-1]}$. If such a pair of paths exists, then there are two possibilities. If $w_{kk}^{[k-1]} > 0$, there are paths of arbitrary long length from $v_i$ to $v_j$, so set $w_{ij}^{[k]} = \infty$. If $w_{kk}^{[k-1]} = 0$, set $w_{ij}^{[k-1]} = \max(w_{ij}^{[k-1]}, w_{ik}^{[k-1]} + w_{kj}^{[k-1]})$. (Initially take $\mathbf{W}_0 = \mathbf{M}_R$.) **21.** 25 **23.** Because $A_i \cap B_j$ is a subset of $A_i$ and of $B_j$, the collection of subsets is a refinement of each of the given partitions. We must show that it is a partition. By construction, each of these sets is nonempty. To see that their union is $S$, suppose that $s \in S$. Because $P_1$ and $P_2$ are partitions of $S$, there are sets $A_i$ and $B_j$ such that $s \in A_i$ and $s \in B_j$. Therefore $s \in A_i \cap B_j$. Hence, the union of these sets is $S$. To see that they are pairwise disjoint, note that unless $i = i'$ and $j = j'$, $(A_i \cap B_j) \cap (A_{i'} \cap B_{j'}) = (A_i \cap A_{i'}) \cap (B_j \cap B_{j'}) = \emptyset$. **25.** The subset relation is a partial ordering on any collection of sets, because it is reflexive, antisymmetric, and transitive. Here the collection of sets is $\mathbf{R}(S)$. **27.** Find recipe $\prec$ Buy seafood $\prec$ Buy groceries $\prec$ Wash shellfish $\prec$ Cut ginger and garlic $\prec$ Clean fish $\prec$ Steam rice $\prec$ Cut fish $\prec$ Wash vegetables $\prec$ Chop water chestnuts $\prec$ Make garnishes $\prec$ Cook in wok $\prec$ Arrange on platter $\prec$ Serve **29. a)** The only antichain with more than one element is $\{c, d\}$. **b)** The only antichains with more than one element are $\{b, c\}$, $\{c, e\}$, and $\{d, e\}$. **c)** The only antichains with more than one element are $\{a, b\}$, $\{a, c\}$, $\{b, c\}$, $\{a, b, c\}$, $\{d, e\}$, $\{d, f\}$, $\{e, f\}$, and $\{d, e, f\}$. **31.** Let $(S, \preccurlyeq)$ be a finite poset, and let $A$ be a maximal chain. Because $(A, \preccurlyeq)$ is also a poset it must have a minimal element $m$. Suppose that $m$ is not minimal in $S$. Then there would be an element $a$ of $S$ with $a \prec m$. However, this would make the set $A \cup \{a\}$ a larger chain than $A$. To show this, we must show that $a$ is comparable with every element of $A$. Because $m$ is comparable with every element of $A$ and $m$ is minimal, it follows that $m \prec x$ when $x$ is in $A$ and $x \neq m$. Because $a \prec m$ and $m \prec x$, the transitive law shows that $a \prec x$ for every element of $A$. **33.** Let $aRb$ denote that $a$ is a descendant of $b$. By Exercise 32, if no set of $n + 1$ people none of whom is a descendant of any other (an antichain) exists, then $k \leq n$, so the set can be partitioned into $k \leq n$ chains. By the pigeonhole principle, at least one of these chains contains at least $m+1$ people. **35.** We prove by contradiction that if $S$ has no infinite decreasing sequence and $\forall x\,(\{\forall y[y \prec x \rightarrow P(y)]\} \rightarrow P(x))$, then $P(x)$ is true for all $x \in S$. If it does not hold that $P(x)$ is true for all $x \in S$, let $x_1$ be an element of $S$ such that $P(x_1)$ is not true. Then by the conditional statement already given, it must be the case that $\forall y[y \prec x_1 \rightarrow P(y)]$ is not true. This means that there is some $x_2$ with $x_2 \prec x_1$ such that $P(x_2)$ is not true. Again invoking the conditional statement, we get an $x_3 \prec x_2$ such that $P(x_3)$ is not true, and so on forever. This contradicts the well-foundedness of our poset. Therefore, $P(x)$ is true for all $x \in S$. **37.** Suppose that $R$ is a quasi-ordering. Because $R$ is reflexive, if $a \in A$, then $(a, a) \in R$. This implies that $(a, a) \in R^{-1}$. Hence, $a \in R \cap R^{-1}$. It follows that $R \cap R^{-1}$ is reflexive. $R \cap R^{-1}$ is symmetric for any relation $R$ because, for any relation $R$, if $(a, b) \in R$ then $(b, a) \in R^{-1}$ and vice versa. To show that $R \cap R^{-1}$ is transitive, suppose that $(a, b) \in R \cap R^{-1}$ and $(b, c) \in R \cap R^{-1}$. Because $(a, b) \in R$ and $(b, c) \in R$, $(a, c) \in R$, because $R$ is transitive. Similarly, because $(a, b) \in R^{-1}$ and $(b, c) \in R^{-1}$, $(b, a) \in R$ and $(c, b) \in R$, so $(c, a) \in R$ and $(a, c) \in R^{-1}$. Hence, $(a, c) \in R \cap R^{-1}$. It follows that $R \cap R^{-1}$ is an equivalence relation. **39. a)** Because $\mathrm{glb}(x, y) = \mathrm{glb}(y, x)$ and $\mathrm{lub}(x, y) = \mathrm{lub}(y, x)$, it follows that $x \wedge y = y \wedge x$ and $x \vee y = y \vee x$. **b)** Using the definition, $(x \wedge y) \wedge z$ is a lower bound of $x$, $y$, and $z$ that is greater than every other lower bound. Because $x$, $y$, and $z$ play interchangeable roles, $x \wedge (y \wedge z)$ is the same element. Similarly, $(x \vee y) \vee z$ is an upper bound of $x$, $y$, and $z$ that is less than every other upper bound. Because $x$, $y$, and $z$ play interchangeable roles, $x \vee (y \vee z)$ is the same element. **c)** To show that $x \wedge (x \vee y) = x$ it is sufficient to show that $x$ is the greatest lower bound of $x$, and $x \vee y$. Note that $x$ is a lower bound of $x$, and because $x \vee y$ is by definition greater than $x$, $x$ is a lower bound for it as well. Therefore, $x$ is a lower bound. But any lower bound of $x$ has to be less than $x$, so $x$ is the greatest lower bound. The second statement is the dual of the first; we omit its proof. **d)** $x$ is a lower, and an upper, bound for itself and itself, and the greatest, and least, such bound. **41. a)** Because 1 is the only element greater than or equal to 1, it is the only upper bound for 1 and therefore the only possible value of the least upper bound of $x$ and 1. **b)** Because $x \preccurlyeq 1$, $x$ is a lower bound for both $x$ and 1 and no other lower bound can be greater than $x$, so $x \wedge 1 = x$. **c)** Because $0 \preccurlyeq x$, $x$ is an upper bound for both $x$ and 0 and no other bound can be less than $x$, so $x \vee 0 = x$. **d)** Because 0 is the only element less than or equal to 0, it is the only lower bound for 0 and therefore the only possible value of the greatest lower bound of $x$ and 0. **43.** $L = (S, \subseteq)$ where $S = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$ **45.** Yes **47.** The complement of a subset $X \subseteq S$ is its complement $S - X$. To prove this, note that $X \vee (S - X) = 1$ and $X \wedge (S - X) = 0$ because $X \cup (S - X) = S$ and $X \cap (S - X) = \emptyset$. **49.** Think of the rectangular grid as representing elements in a matrix. Thus we number from top to bottom and within that from left to right. The partial order is that $(a, b) \preceq (c, d)$ iff $a \leq c$ and $b \leq d$. Note that $(1, 1)$ is the least element under this relation. The rules for Chomp as explained in Chapter 1 coincide with the rules stated in the preamble here. But now we can identify the point $(a, b)$ with the natural number $p^{a-1}q^{b-1}$ for all $a$ and $b$ with $1 \leq a \leq m$ and $1 \leq b \leq n$. This identifies the points in the rectangular grid with the set $S$ in this exercise, and the partial order $\preceq$ just described is the same as the divides relation, because $p^{a-1}q^{b-1} \mid p^{c-1}q^{d-1}$ if and only if the exponent of $p$ on the left does not exceed the exponent of $p$ on the right, and similarly for $q$.

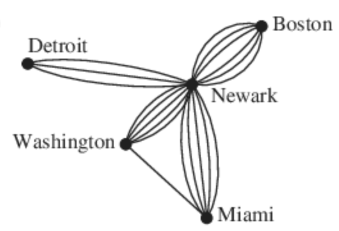
## CHAPTER 10

### Section 10.1

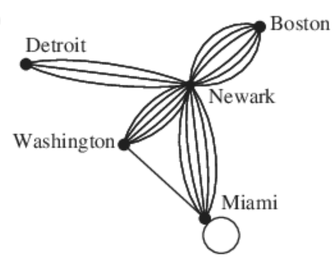
**1. a)**



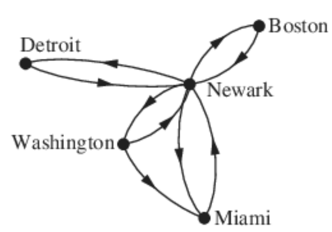

**b)**



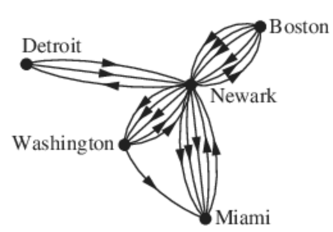
**c)**




**d)**

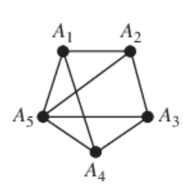



**e)**

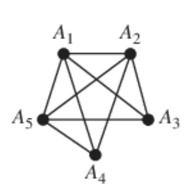



**3.** Simple graph **5.** Pseudograph **7.** Directed graph **9.** Directed multigraph **11.** If $uRv$, then there is an edge associated with $\{u, v\}$. But $\{u, v\} = \{v, u\}$, so this edge is associated with $\{v, u\}$ and therefore $vRu$. Thus, by definition, $R$ is a symmetric relation. A simple graph does not allow loops; therefore, $uRu$ never holds, and so by definition $R$ is irreflexive.

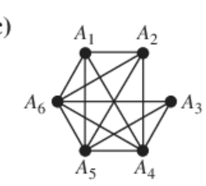
**13. a)**



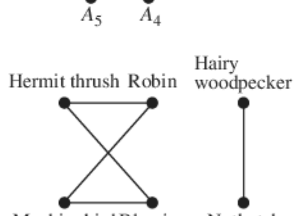

**b)**



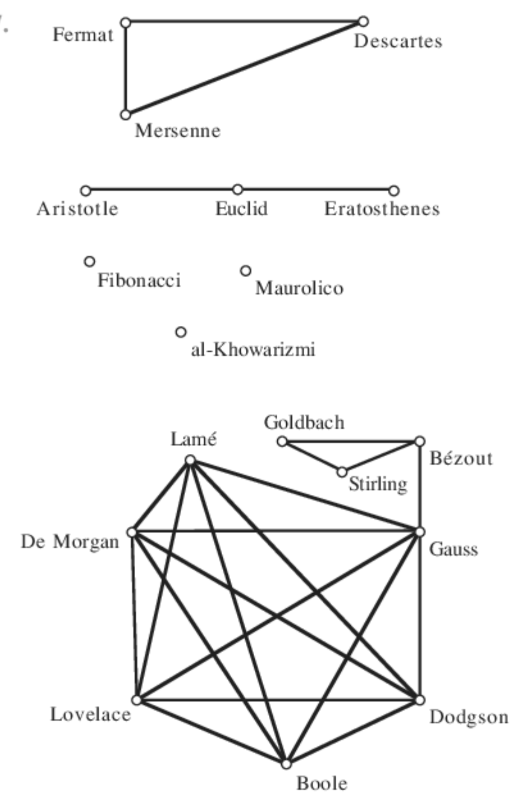

**c)**



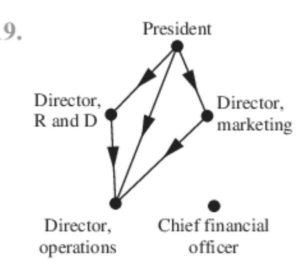

**15.**



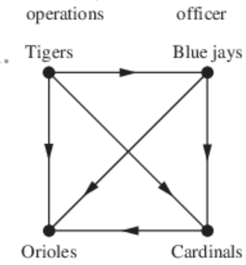

**17.**




**19.**

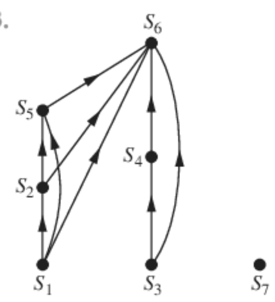



**21.**




**23.** We find the telephone numbers in the call graph for February that are not present in the call graph for January and vice versa. For each number we find, we make a list of the numbers they called or were called by using the edges in the call graph. We examine these lists to find new telephone numbers in February that had similar calling patterns to defunct telephone numbers in January. **25.** We use the graph model that has e-mail addresses as vertices and for each message sent, an edge from the e-mail address of the sender to the e-mail address of the recipient. For each e-mail address, we can make a list of other addresses they sent messages to and a list of other addresses from which they received messages. If two e-mail addresses had almost the same pattern, we conclude that these addresses might have belonged to the same person who had recently changed his or her e-mail address. **27.** Let $V$ be the set of people at the party. Let $E$ be the set of ordered pairs $(u, v)$ in $V \times V$ such that $u$ knows $v$'s name. The edges are directed, but multiple edges are not allowed. Literally, there is a loop at each vertex, but for simplicity, the model could omit the loops. **29.** Vertices are the courses; edges are directed; edge $uv$ means that course $u$ is prerequisite for course $v$; courses without prerequisites are vertices with in-degree 0; courses that are not prerequisite for any other courses are vertices with out-degree 0. **31.** Let the set of vertices be a set of people, and two vertices are joined by an edge if the two people were ever married. Ignoring complications, this graph has the property that there are two types of vertices (men and women), and every edge joins vertices of opposite types.

**33.**

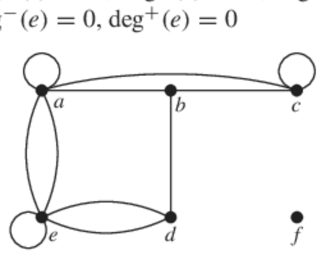



**35.** Represent people in the group by vertices. Put a directed edge into the graph for every pair of vertices. Label the edge from the vertex representing $A$ to the vertex representing $B$ with a + (plus) if $A$ likes $B$, a − (minus) if $A$ dislikes $B$, and a 0 if $A$ is neutral about $B$.

## Section 10.2

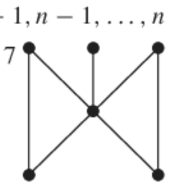
**1.** $v = 6$; $e = 6$; $\deg(a) = 2$, $\deg(b) = 4$, $\deg(c) = 1$, $\deg(d) = 0$, $\deg(e) = 2$, $\deg(f) = 3$; $c$ is pendant; $d$ is isolated. **3.** $v = 9$; $e = 12$; $\deg(a) = 3$, $\deg(b) = 2$, $\deg(c) = 4$, $\deg(d) = 0$, $\deg(e) = 6$, $\deg(f) = 0$; $\deg(g) = 4$; $\deg(h) = 2$; $\deg(i) = 3$; $d$ and $f$ are isolated. **5.** No **7.** $v = 4$; $e = 7$; $\deg^-(a) = 3$, $\deg^-(b) = 1$, $\deg^-(c) = 2$, $\deg^-(d) = 1$, $\deg^+(a) = 1$, $\deg^+(b) = 2$, $\deg^+(c) = 1$, $\deg^+(d) = 3$ **9.** 5 vertices, 13 edges; $\deg^-(a) = 6$, $\deg^+(a) = 1$, $\deg^-(b) = 1$, $\deg^+(b) = 5$, $\deg^-(c) = 2$, $\deg^+(c) = 5$, $\deg^-(d) = 4$, $\deg^+(d) = 2$, $\deg^-(e) = 0$, $\deg^+(e) = 0$

**11.**




**13.** The number of coauthors that person has; that person's coauthors; a person who has no coauthors; a person who has only one coauthor **15.** In the directed graph $\deg^-(v) =$ number of calls $v$ received, $\deg^+(v) =$ number of calls $v$ made; in the undirected graph, $\deg(v)$ is the number of calls either made or received by $v$. **17.** $(\deg^+(v), \deg^-(v))$ is the win–loss record of $v$. **19.** In the undirected graph model in which the vertices are people in the group and two vertices are adjacent if those two people are friends, the degree of a vertex is the number of friends in the group that person has. By Exercise 18, there are two vertices with the same degree, which means that there are two people in the group with the same number of friends in the group. **21.** Bipartite **23.** Not bipartite **25.** Not bipartite **27. a)** Parts $\{h, s, n, w\}$ and $\{P, Q, R, S\}$, $E = \{\{P, n\}, \{P, w\}, \{Q, s\}, \{Q, n\}, \{R, n\}, \{R, w\}, \{S, h\}, \{S, s\}\}$ **b)** There is. **c)** $\{Pw, Qs, Rn, Sh\}$ among others **29.** Only Barry is willing to marry Uma and Xia. **31.** Model this with an undirected bipartite graph, with an edge between a man and a woman if they are willing to marry each other. By Hall's theorem, it is enough to show that for every set $S$ of women, the set $N(S)$ of men willing to marry them has cardinality at least $|S|$. Let $m$ be the number of edges between $S$ and $N(S)$. Since every vertex in $S$ has degree $k$, it follows that $m = k|S|$. Because these edges are incident to $N(S)$, it follows that $m \leq k|N(S)|$. Therefore $k|S| \leq k|N(S)|$, so $|N(S)| \geq |S|$. **33. a)** $(\{a, b, c, f\}, \{\{a, b\}, \{a, f\}, \{b, c\}, \{b, f\}\})$ **b)** $(\{a, x, c, f\}, \{\{a, x\}, \{c, x\}, \{e, x\}\})$ **35. a)** $n$ vertices, $n(n-1)/2$ edges **b)** $n$ vertices, $n$ edges **c)** $n+1$ vertices, $2n$ edges **d)** $m+n$ vertices, $mn$ edges **e)** $2^n$ vertices, $n2^{n-1}$ edges **37. a)** 3, 3, 3, 3 **b)** 2, 2, 2, 2 **c)** 4, 3, 3, 3, 3 **d)** 3, 3, 2, 2, 2 **e)** 3, 3, 3, 3, 3, 3, 3, 3 **39.** Each of the $n$ vertices is adjacent to each of the other $n - 1$ vertices, so the degree sequence is $n-1, n-1, \ldots, n-1$ ($n$ terms).

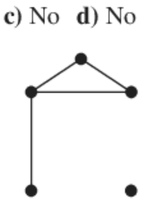
**41.** 7



**43. a)** Yes

**b)** No **c)** No **d)** No

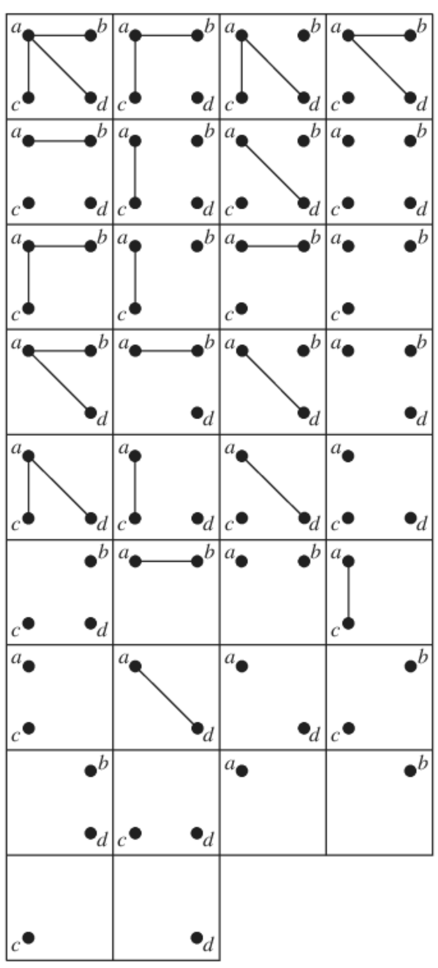
**e)** Yes



**f)** No. **45.** First, suppose that $d_1, d_2, \ldots, d_n$ is graphic. We must show that the sequence whose terms are $d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, d_{d_1+3}, \ldots, d_n$ is graphic once it is put into nonincreasing order. In Exercise 44 it is proved that if the original sequence is graphic, then in fact there is a graph having this degree sequence in which the vertex of degree $d_1$ is adjacent to the vertices of degrees $d_2, d_3, \ldots, d_{d_1+1}$. Remove from this graph the vertex of highest degree ($d_1$). The resulting graph has the desired degree sequence. Conversely, suppose that $d_1, d_2, \ldots, d_n$ is a nonincreasing sequence such that the sequence

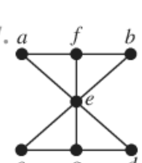
$d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, d_{d_1+3}, \ldots, d_n$ is graphic once it is put into nonincreasing order. Take a graph with this latter degree sequence, where vertex $v_i$ has degree $d_i - 1$ for $2 \leq i \leq d_1 + 1$ and vertex $v_i$ has degree $d_i$ for $d_1 + 2 \leq i \leq n$. Adjoin one new vertex (call it $v_1$), and put in an edge from $v_1$ to each of the vertices $v_2, v_3, \ldots, v_{d_1+1}$. The resulting graph has degree sequence $d_1, d_2, \ldots, d_n$. **47.** Let $d_1, d_2, \ldots, d_n$ be a nonincreasing sequence of nonnegative integers with an even sum. Construct a graph as follows: Take vertices $v_1, v_2, \ldots, v_n$ and put $\lfloor d_i/2 \rfloor$ loops at vertex $v_i$, for $i = 1, 2, \ldots, n$. For each $i$, vertex $v_i$ now has degree either $d_i$ or $d_i - 1$. Because the original sum was even, the number of vertices for which $\deg(v_i) = d_i - 1$ is even. Pair them up arbitrarily, and put in an edge joining the vertices in each pair. **49.** 17

**51.**



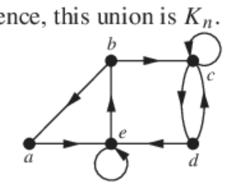
**53. a)** For all $n \geq 1$ **b)** For all $n \geq 3$ **c)** For $n = 3$ **d)** For all $n \geq 0$ **55.** 5

**57.**

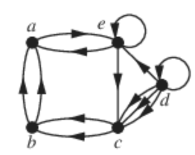



**59. a)** The graph with $n$ vertices and no edges **b)** The disjoint union of $K_m$ and $K_n$ **c)** The graph with vertices $\{v_1, \ldots, v_n\}$ with an edge between $v_i$ and $v_j$ unless $i \equiv j \pm 1 \pmod{n}$ **d)** The graph whose vertices are represented by bit strings of length $n$ with an edge between two vertices if the associated bit strings differ in more than one bit **61.** $v(v-1)/2 - e$ **63.** $n - 1 - d_n, n - 1 - d_{n-1}, \ldots, n - 1 - d_2, n - 1 - d_1$ **65.** The union of $G$ and $\overline{G}$ contains an edge between each pair of the $n$ vertices. Hence, this union is $K_n$.

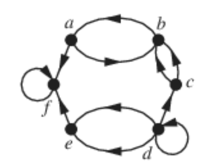
**67.** Exercise 7:



Exercise 8:

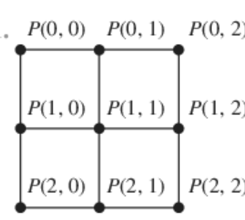


Exercise 9:



**69.** A directed graph $G = (V, E)$ is its own converse if and only if it satisfies the condition $(u, v) \in E$ if and only if $(v, u) \in E$. But this is precisely the condition that the associated relation must satisfy to be symmetric.

**71.**




**73.** We can connect $P(i, j)$ and $P(k, l)$ by using $|i - k|$ hops to connect $P(i, j)$ and $P(k, j)$ and $|j - l|$ hops to connect $P(k, j)$ and $P(k, l)$. Hence, the total number of hops required to connect $P(i, j)$ and $P(k, l)$ does not exceed $|i - k| + |j - l|$. This is less than or equal to $m + m = 2m$, which is $O(m)$.

## Section 10.3

**1.**

| *Vertex* | *Adjacent Vertices* |
|---|---|
| $a$ | $b, c, d$ |
| $b$ | $a, d$ |
| $c$ | $a, d$ |
| $d$ | $a, b, c$ |

**3.**

| *Vertex* | *Terminal Vertices* |
|---|---|
| $a$ | $a, b, c, d$ |
| $b$ | $d$ |
| $c$ | $a, b$ |
| $d$ | $b, c, d$ |

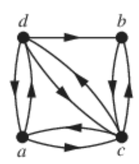
**5.** $\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$

**7.** $\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$ **9. a)** $\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$

**b)** $\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$ **c)** $\begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}$

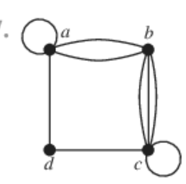
**d)** $\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ **e)** $\begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$

**f)** $\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$

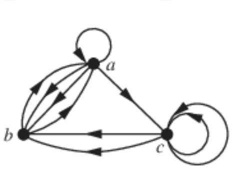
**11.** 


**13.** $\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 \\ 0 & 2 & 1 & 0 \end{bmatrix}$

**15.** $\begin{bmatrix} 1 & 0 & 2 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}$

**17.** 


**19.** $\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$ **21.** $\begin{bmatrix} 1 & 1 & 2 & 1 \\ 1 & 0 & 0 & 2 \\ 1 & 0 & 1 & 1 \\ 0 & 2 & 1 & 0 \end{bmatrix}$

**23.** 


**25.** Yes

**27.** Exercise 13: $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$

Exercise 14: $\begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$

Exercise 15: $\begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

**29.** $\deg(v)$ − number of loops at $v$; $\deg^-(v)$ **31.** 2 if $e$ is not a loop, 1 if $e$ is a loop

**33. a)** $\begin{bmatrix} 1 & 1 & \cdots & 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 1 \end{bmatrix}$

**b)** $\begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 1 & 1 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}$

**c)** $\begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\ & & & & 1 & 0 & \cdots & 0 \\ & & \mathbf{B} & & 0 & 1 & \cdots & 0 \\ & & & & \vdots & \vdots & & \vdots \\ & & & & 0 & 0 & \cdots & 1 \end{bmatrix}$

where **B** is the answer to (b)

**d)** $\begin{bmatrix} 1 & 1 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & \cdots & 1 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \end{bmatrix}$

**35.** Isomorphic **37.** Isomorphic **39.** Isomorphic **41.** Not isomorphic **43.** Isomorphic **45.** $G$ is isomorphic to itself by the identity function, so isomorphism is reflexive. Suppose that $G$ is isomorphic to $H$. Then there exists a one-to-one correspondence $f$ from $G$ to $H$ that preserves adjacency and nonadjacency. It follows that $f^{-1}$ is a one-to-one correspondence from $H$ to $G$ that preserves adjacency and nonadjacency. Hence, isomorphism is symmetric. If $G$ is isomorphic to $H$ and $H$ is isomorphic to $K$, then there are one-to-one correspondences $f$ and $g$ from $G$ to $H$ and from $H$ to $K$ that preserve adjacency and nonadjacency. It follows that $g \circ f$ is a one-to-one correspondence from $G$ to $K$ that preserves adjacency and nonadjacency. Hence, isomorphism is transitive. **47.** All zeros **49.** Label the vertices in order so that all of the vertices in the first set of the partition of the vertex set come first. Because no edges join vertices in the same set of the partition, the matrix has the desired form. **51.** $C_5$ **53.** $n = 5$ only **55.** 4 **57. a)** Yes **b)** No **c)** No **59.** $G = (V_1, E_1)$ is isomorphic to $H = (V_2, E_2)$ if and only if there exist functions $f$ from $V_1$ to $V_2$ and $g$ from $E_1$ to $E_2$ such that each is a one-to-one correspondence and for every edge $e$ in $E_1$ the endpoints of $g(e)$ are $f(v)$ and $f(w)$ where $v$ and $w$ are the endpoints of $e$. 61. Yes 63. Yes 65. If $f$ is an isomorphism from a directed graph $G$ to a directed graph $H$, then $f$ is also an isomorphism from $G^{conv}$ to $H^{conv}$. To see this note that $(u, v)$ is an edge of $G^{conv}$ if and only if $(v, u)$ is an edge of $G$ if and only if $(f(v), f(u))$ is an edge of $H$ if and only if $(f(u), f(v))$ is an edge of $H^{conv}$. 67. Many answers are possible; for example, $C_6$ and $C_3 \cup C_3$. 69. The product is $[a_{ij}]$ where $a_{ij}$ is the number of edges from $v_i$ to $v_j$ when $i \neq j$ and $a_{ii}$ is the number of edges incident to $v_i$. 71. The graphs in Exercise 41 provide a devil's pair.

## Section 10.4

1. **a)** Path of length 4; not a circuit; not simple **b)** Not a path **c)** Not a path **d)** Simple circuit of length 5 3. No 5. No 7. Maximal sets of people with the property that for any two of them, we can find a string of acquaintances that takes us from one to the other 9. If a person has Erdős number $n$, then there is a path of length $n$ from that person to Erdős in the collaboration graph, so by definition, that means that that person is in the same component as Erdős. If a person is in the same component as Erdős, then there is a path from that person to Erdős, and the length of the shortest such path is that person's Erdős number. 11. **a)** Weakly connected **b)** Weakly connected **c)** Not strongly or weakly connected 13. The maximal sets of phone numbers for which it is possible to find directed paths between every two different numbers in the set 15. **a)** $\{a, b, f\}$, $\{c, d, e\}$ **b)** $\{a, b, c, d, e, h\}$, $\{f\}$, $\{g\}$ **c)** $\{a, b, d, e, f, g, h, i\}$, $\{c\}$ 17. Suppose the strong components of $u$ and $v$ are not disjoint, say with vertex $w$ in both. Suppose $x$ is a vertex in the strong component of $u$. Then $x$ is also in the strong component of $v$, because there is a path from $x$ to $v$ (namely the path from $x$ to $u$ followed by the path from $u$ to $w$ followed by the path from $w$ to $v$) and vice versa. Thus $x$ is in the strong component of $v$. This shows that the strong component of $u$ is a subgraph of the strong component of $v$, and equality follows by symmetry. 19. **a)** 2 **b)** 7 **c)** 20 **d)** 61 21. Not isomorphic ($G$ has a triangle; $H$ does not) 23. Isomorphic (the path $u_1, u_2, u_7, u_6, u_5, u_4, u_3, u_8, u_1$ corresponds to the path $v_1, v_2, v_3, v_4, v_5, v_8, v_7, v_6, v_1$) 25. **a)** 3 **b)** 0 **c)** 27 **d)** 0 27. **a)** 1 **b)** 0 **c)** 2 **d)** 1 **e)** 5 **f)** 3 29. $R$ is reflexive by definition. Assume that $(u, v) \in R$; then there is a path from $u$ to $v$. Then $(v, u) \in R$ because there is a path from $v$ to $u$, namely, the path from $u$ to $v$ traversed backward. Assume that $(u, v) \in R$ and $(v, w) \in R$; then there are paths from $u$ to $v$ and from $v$ to $w$. Putting these two paths together gives a path from $u$ to $w$. Hence, $(u, w) \in R$. It follows that $R$ is transitive. 31. $c$ 33. $b, c, e, i$ 35. If a vertex is pendant it is clearly not a cut vertex. So an endpoint of a cut edge that is a cut vertex is not pendant. Removal of a cut edge produces a graph with more connected components than in the original graph. If an endpoint of a cut edge is not pendant, the connected component it is in after the removal of the cut edge contains more than just this vertex. Consequently, removal of that vertex and all edges incident to it, including the original cut edge, produces a graph with more connected components than were in the original graph. Hence, an endpoint of a cut edge that is not pendant is a cut vertex. 37. Assume there exists a connected graph $G$ with at most one vertex that is not a cut vertex. Define the distance between the vertices $u$ and $v$, denoted by $d(u, v)$, to be the length of the shortest path between $u$ and $v$ in $G$. Let $s$ and $t$ be vertices in $G$ such that $d(s, t)$ is a maximum. Either $s$ or $t$ (or both) is a cut vertex, so without loss of generality suppose that $s$ is a cut vertex. Let $w$ belong to the connected component that does not contain $t$ of the graph obtained by deleting $s$ and all edges incident to $s$ from $G$. Because every path from $w$ to $t$ contains $s$, $d(w, t) > d(s, t)$, which is a contradiction. 39. **a)** Denver–Chicago, Boston–New York **b)** Seattle–Portland, Portland–San Francisco, Salt Lake City–Denver, New York–Boston, Boston–Burlington, Boston–Bangor 41. A minimal set of people who collectively influence everyone (directly or indirectly); {Deborah} 43. An edge cannot connect two vertices in different connected components. Because there are at most $C(n_i, 2)$ edges in the connected component with $n_i$ vertices, it follows that there are at most $\sum_{i=1}^{k} C(n_i, 2)$ edges in the graph. 45. Suppose that $G$ is not connected. Then it has a component of $k$ vertices for some $k$, $1 \leq k \leq n-1$. The most edges $G$ could have is $C(k, 2) + C(n-k, 2) = [k(k-1) + (n-k)(n-k-1)]/2 = k^2 - nk + (n^2 - n)/2$. This quadratic function of $f$ is minimized at $k = n/2$ and maximized at $k = 1$ or $k = n-1$. Hence, if $G$ is not connected, the number of edges does not exceed the value of this function at 1 and at $n-1$, namely, $(n-1)(n-2)/2$. 47. **a)** 1 **b)** 2 **c)** 6 **d)** 21 49. **a)** Removing an edge from a cycle leaves a path, which is still connected. **b)** Removing an edge from the cycle portion of the wheel leaves that portion still connected and the central vertex still connected to it as well. Removing a spoke leaves the cycle intact and the central vertex still connected to it as well. **c)** Any four vertices, two from each part of the bipartition, are connected by a 4-cycle; removing one edge does not disconnect them. **d)** Deleting the edge joining $(b_1, b_2, \ldots, b_{i-1}, 0, b_{i+1}, \ldots, b_n)$ and $(b_1, b_2, \ldots, b_{i-1}, 1, b_{i+1}, \ldots, b_n)$ does not disconnect the graph because these two vertices are still joined via the path $(b_1, b_2, \ldots, b_{i-1}, 0, b_{i+1}, \ldots, 0)$, $(b_1, b_2, \ldots, b_{i-1}, 0, b_{i+1}, \ldots, 1)$, $(b_1, b_2, \ldots, b_{i-1}, 1, b_{i+1}, \ldots, 1)$, $(b_1, b_2, \ldots, b_{i-1}, 1, b_{i+1}, \ldots, 0)$ if $n < 2$ and $b_n = 0$, and similarly in the other three cases. 51. If $G$ is complete, then removing vertices one by one leaves a complete graph at each step, so we never get a disconnected graph. Conversely, if edge $uv$ is missing from $G$, then removing all the vertices except $u$ and $v$ creates a disconnected graph. 53. Both equal $\min(m, n)$. 55. Let $G$ be a graph with $n$ vertices; then $\kappa(G) \leq n-1$. Let $C$ be a smallest edge cut, leaving a nonempty proper subset $S$ of the vertices of $G$ disconnected from the complementary set $S' = V - S$. If $xy$ is an edge of $G$ for every $x \in S$ and $y \in S'$, then the size of $C$ is $|S||S'|$, which is at least $n-1$, so $\kappa(G) \leq \lambda(G)$. Otherwise, let $x \in S$ and $y \in S'$ be nonadjacent vertices. Let $T$ consist of all neighbors of $x$ in $S'$ together with all vertices of $S - \{x\}$ with neighbors

in $S'$. Then $T$ is a vertex cut, because it separates $x$ and $y$. Now look at the edges from $x$ to $T \cap S'$ and one edge from each vertex of $T \cap S$ to $S'$; this gives us $|T|$ distinct edges that lie in $C$, so $\lambda(G) = |C| \geq |T| \geq \kappa(G)$. **57.** 2 **59.** Let the simple paths $P_1$ and $P_2$ be $u = x_0, x_1, \ldots, x_n = v$ and $u = y_0, y_1, \ldots, y_m = v$, respectively. The paths thus start out at the same vertex. Since the paths do not contain the same set of edges, they must diverge eventually. If they diverge only after one of them has ended, then the rest of the other path is a simple circuit from $v$ to $v$. Otherwise we can suppose that $x_0 = y_0, x_1 = y_1, \ldots, x_i = y_i$, but $x_{i+1} \neq y_{i+1}$. To form our simple circuit, we follow the path $y_i, y_{i+1}, y_{i+2}$, and so on, until it once again first encounters a vertex on $P_1$ (possibly as early as $y_{i+1}$, no later than $y_m$). Once we are back on $P_1$, we follow it along—forwards or backwards, as necessary—to return to $x_i$. Since $x_i = y_i$, this certainly forms a circuit. It must be a simple circuit, since no edge among the $x_k$s or the $y_l$s can be repeated ($P_1$ and $P_2$ are simple by hypothesis) and no edge among the $x_k$s can equal one of the edges $y_l$ that we used, since we abandoned $P_2$ for $P_1$ as soon as we hit $P_1$. **61.** The graph $G$ is connected if and only if every off-diagonal entry of $\mathbf{A}+\mathbf{A}^2+\mathbf{A}^3+\cdots+\mathbf{A}^{n-1}$ is positive, where $\mathbf{A}$ is the adjacency matrix of $G$. **63.** If the graph is bipartite, say with parts $A$ and $B$, then the vertices in every path must alternately lie in $A$ and $B$. Therefore a path that starts in $A$, say, will end in $B$ after an odd number of steps and in $A$ after an even number of steps. Because a circuit ends at the same vertex where it starts, the length must be even. Conversely, suppose that all circuits have even length; we must show that the graph is bipartite. We can assume that the graph is connected, because if it is not, then we can just work on one component at a time. Let $v$ be a vertex of the graph, and let $A$ be the set of all vertices to which there is a path of odd length starting at $v$, and let $B$ be the set of all vertices to which there is a path of even length starting at $v$. Because the component is connected, every vertex lies in $A$ or $B$. No vertex can lie in both $A$ and $B$, because if one did, then following the odd-length path from $v$ to that vertex and then back along the even-length path from that vertex to $v$ would produce an odd circuit, contrary to the hypothesis. Thus, the set of vertices has been partitioned into two sets. To show that every edge has endpoints in different parts, suppose that $xy$ is an edge, where $x \in A$. Then the odd-length path from $v$ to $x$ followed by $xy$ produces an even-length path from $v$ to $y$, so $y \in B$. (Similarly, if $x \in B$.) **65.** $(H_1W_1H_2W_2\langle\text{boat}\rangle, \emptyset) \to (H_2W_2, H_1W_1\langle\text{boat}\rangle) \to (H_1H_2W_2\langle\text{boat}\rangle, W_1) \to (W_2, H_1W_1H_2\langle\text{boat}\rangle) \to (H_2W_2\langle\text{boat}\rangle, H_1W_1) \to (\emptyset, H_1W_1H_2W_2\langle\text{boat}\rangle)$

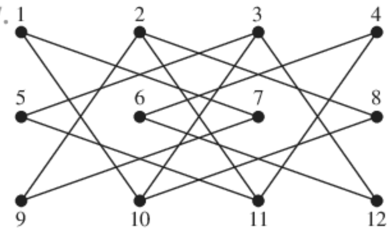
## Section 10.5

**1.** Neither **3.** No Euler circuit; $a, e, c, e, b, e, d, b, a, c, d$ **5.** $a, b, c, d, c, e, d, b, e, a, e, a$ **7.** $a, i, h, g, d, e, f, g, c, e, h, d, c, a, b, i, c, b, h, a$ **9.** No, $A$ still has odd degree. **11.** When the graph in which vertices represent intersections and edges streets has an Euler path **13.** Yes **15.** No **17.** If there is an Euler path, then as we follow it each vertex except the starting and ending vertices must have equal in-degree and out-degree, because whenever we come to a vertex along an edge, we leave it along another edge. The starting vertex must have out-degree 1 larger than its in-degree, because we use one edge leading out of this vertex and whenever we visit it again we use one edge leading into it and one leaving it. Similarly, the ending vertex must have in-degree 1 greater than its out-degree. Because the Euler path with directions erased produces a path between any two vertices, in the underlying undirected graph, the graph is weakly connected. Conversely, suppose the graph meets the degree conditions stated. If we add one more edge from the vertex of deficient out-degree to the vertex of deficient in-degree, then the graph has every vertex with equal in-degree and out-degree. Because the graph is still weakly connected, by Exercise 16 this new graph has an Euler circuit. Now delete the added edge to obtain the Euler path. **19.** Neither **21.** No Euler circuit; $a, d, e, d, b, a, e, c, e, b, c, b, e$ **23.** Neither **25.** Follow the same procedure as Algorithm 1, taking care to follow the directions of edges. **27.** **a)** $n = 2$ **b)** None **c)** None **d)** $n = 1$ **29.** Exercise 1:1 time; Exercises 2–7: 0 times **31.** $a, b, c, d, e, a$ is a Hamilton circuit. **33.** No Hamilton circuit exists, because once a purported circuit has reached $e$ it would have nowhere to go. **35.** No Hamilton circuit exists, because every edge in the graph is incident to a vertex of degree 2 and therefore must be in the circuit. **37.** $a, b, c, f, d, e$ is a Hamilton path. **39.** $f, e, d, a, b, c$ is a Hamilton path. **41.** No Hamilton path exists. There are eight vertices of degree 2, and only two of them can be end vertices of a path. For each of the other six, their two incident edges must be in the path. It is not hard to see that if there is to be a Hamilton path, exactly one of the inside corner vertices must be an end, and that this is impossible. **43.** $a, b, c, f, i, h, g, d, e$ is a Hamilton path. **45.** $m = n \geq 2$ **47.** **a)** (i) No, (ii) No, (iii) Yes **b)** (i) No, (ii) No, (iii) Yes **c)** (i) Yes, (ii) Yes, (iii) Yes **d)** (i) Yes, (ii) Yes, (iii) Yes **49.** The result is trivial for $n = 1$: code is 0, 1. Assume we have a Gray code of order $n$. Let $c_1, \ldots, c_k$, $k = 2^n$ be such a code. Then $0c_1, \ldots, 0c_k, 1c_k, \ldots, 1c_1$ is a Gray code of order $n + 1$.

**51.** **procedure** *Fleury*($G = (V, E)$: connected multigraph with the degrees of all vertices even, $V = \{v_1, \ldots, v_n\}$)

```
v := v_1
circuit := v
H := G
while H has edges
    e := first edge with endpoint v in H (with respect
    to listing of V) such that e is not a cut edge of H, if
    one exists, and simply the first edge in H with
    endpoint v otherwise
    w := other endpoint of e
    circuit := circuit with e, w added
    v := w
    H := H − e
return circuit {circuit is an Euler circuit}
```

53. If $G$ has an Euler circuit, then it also has an Euler path. If not, add an edge between the two vertices of odd degree and apply the algorithm to get an Euler circuit. Then delete the new edge. 55. Suppose $G = (V, E)$ is a bipartite graph with $V = V_1 \cup V_2$, where $V_1 \cap V_2 = \emptyset$ and no edge connects a vertex in $V_1$ and a vertex in $V_2$. Suppose that $G$ has a Hamilton circuit. Such a circuit must be of the form $a_1, b_1, a_2, b_2, \ldots, a_k, b_k, a_1$, where $a_i \in V_1$ and $b_i \in V_2$ for $i = 1, 2, \ldots, k$. Because the Hamilton circuit visits each vertex exactly once, except for $v_1$, where it begins and ends, the number of vertices in the graph equals $2k$, an even number. Hence, a bipartite graph with an odd number of vertices cannot have a Hamilton circuit.

57.



59. We represent the squares of a $3 \times 4$ chessboard as follows:

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |

A knight's tour can be made by following the moves 8, 10, 1, 7, 9, 2, 11, 5, 3, 12, 6, 4. 61. We represent the squares of a $4 \times 4$ chessboard as follows:

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 |

There are only two moves from each of the four corner squares. If we include all the edges 1–10, 1–7, 16–10, and 16–7, a circuit is completed too soon, so at least one of these edges must be missing. Without loss of generality, assume the path starts 1–10, 10–16, 16–7. Now the only moves from square 3 are to squares 5, 10, and 12, and square 10 already has two incident edges. Therefore, 3–5 and 3–12 must be in the Hamilton circuit. Similarly, edges 8–2 and 8–15 must be in the circuit. Now the only moves from square 9 are to squares 2, 7, and 15. If there were edges from square 9 to both squares 2 and 15, a circuit would be completed too soon. Therefore the edge 9–7 must be in the circuit giving square 7 its full complement of edges. But now square 14 is forced to be joined to squares 5 and 12, completing a circuit too soon (5–14–12–3–5). This contradiction shows that there is no knight's tour on the $4 \times 4$ board. 63. Because there are $mn$ squares on an $m \times n$ board, if both $m$ and $n$ are odd, there are an odd number of squares. Because by Exercise 62 the corresponding graph is bipartite, by Exercise 55 it has no Hamilton circuit. Hence, there is no reentrant knight's tour. 65. **a)** If $G$ does not have a Hamilton circuit, continue as long as possible adding missing edges one at a time in such a way that we do not obtain a graph with a Hamilton circuit. This cannot go on forever, because once we've formed the complete graph by adding all missing edges, there is a Hamilton circuit. Whenever the process stops, we have obtained a (necessarily noncomplete) graph $H$ with the desired property. **b)** Add one more edge to $H$. This produces a Hamilton circuit, which uses the added edge. The path consisting of this circuit with the added edge omitted is a Hamilton path in $H$. **c)** Clearly $v_1$ and $v_n$ are not adjacent in $H$, because $H$ has no Hamilton circuit. Therefore they are not adjacent in $G$. But the hypothesis was that the sum of the degrees of vertices not adjacent in $G$ was at least $n$. This inequality can be rewritten as $n - \deg(v_n) \le \deg(v_1)$. But $n - \deg(v_n)$ is just the number of vertices not adjacent to $v_n$. **d)** Because there is no vertex following $v_n$ in the Hamilton path, $v_n$ is not in $S$. Each one of the $\deg(v_1)$ vertices adjacent to $v_1$ gives rise to an element of $S$, so $S$ contains $\deg(v_1)$ vertices. **e)** By part (c) there are at most $\deg(v_1) - 1$ vertices other than $v_n$ not adjacent to $v_n$, and by part (d) there are $\deg(v_1)$ vertices in $S$, none of which is $v_n$. Therefore at least one vertex of $S$ is adjacent to $v_n$. By definition, if $v_k$ is this vertex, then $H$ contains edges $v_k v_n$ and $v_1 v_{k+1}$, where $1 < k < n-1$. **f)** Now $v_1, v_2, \ldots, v_{k-1}, v_k, v_n, v_{n-1}, \ldots, v_{k+1}, v_1$ is a Hamilton circuit in $H$, contradicting the construction of $H$. Therefore, our assumption that $G$ did not originally have a Hamilton circuit is wrong, and our proof by contradiction is complete.

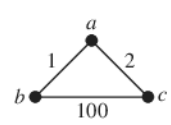
## Section 10.6

1. **a)** Vertices are the stops, edges join adjacent stops, weights are the times required to travel between adjacent stops. **b)** Same as part (a), except weights are distances between adjacent stops. **c)** Same as part (a), except weights are fares between stops. 3. 16 5. Exercise 2: *a, b, e, d, z*; Exercise 3: *a, c, d, e, g, z*; Exercise 4: *a, b, e, h, l, m, p, s, z* 7. **a)** *a, c, d* **b)** *a, c, d, f* **c)** *c, d, f* **e)** *b, d, e, g, z* 9. **a)** Direct **b)** Via New York **c)** Via Atlanta and Chicago **d)** Via New York 11. **a)** Via Chicago **b)** Via Chicago **c)** Via Los Angeles **d)** Via Chicago 13. **a)** Via Chicago **b)** Via Chicago **c)** Via Los Angeles **d)** Via Chicago 15. Do not stop the algorithm when $z$ is added to the set $S$. 17. **a)** Via Woodbridge, via Woodbridge and Camden **b)** Via Woodbridge, via Woodbridge and Camden 19. For instance, sightseeing tours, street cleaning

**21.**

| | $a$ | $b$ | $c$ | $d$ | $e$ | $z$ |
|---|---|---|---|---|---|---|
| $a$ | 4 | 3 | 2 | 8 | 10 | 13 |
| $b$ | 3 | 2 | 1 | 5 | 7 | 10 |
| $c$ | 2 | 1 | 2 | 6 | 8 | 11 |
| $d$ | 8 | 5 | 6 | 4 | 2 | 5 |
| $e$ | 10 | 7 | 8 | 2 | 4 | 3 |
| $z$ | 13 | 10 | 11 | 5 | 3 | 6 |

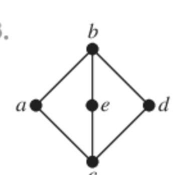
**23.** $O(n^3)$ **25.** $a$–$c$–$b$–$d$–$a$ (or the same circuit starting at some other point and/or traversing the vertices in reverse order) **27.** San Francisco–Denver–Detroit–New York–Los Angeles–San Francisco (or the same circuit starting at some other point and/or traversing the vertices in reverse order) **29.** Consider this graph:




The circuit $a$-$b$-$a$-$c$-$a$ visits each vertex at least once (and the vertex $a$ twice) and has total weight 6. Every Hamilton circuit has total weight 103. **31.** Let $v_1, v_2, \ldots, v_n$ be a topological ordering of the vertices of the given directed acyclic graph. Let $w(i, j)$ be the weight of edge $v_i v_j$. Iteratively define $P(i)$ with the intent that it will be the weight of a longest path ending at $v_i$ and $C(i)$ with the intent that it will be the vertex preceding $v_i$ in some longest path: For $i$ from 1 to $n$, let $P(i)$ be the maximum of $P(j) + w(j, i)$ over all $j < i$ such that $v_j v_i$ is an edge in the directed graph (and if such a $j$ exists let $C(i)$ be a value of $j$ for which this maximum is achieved) and let $P(i) = 0$ if there are no such values of $j$. At the conclusion of this loop, a longest path can be found by choosing $i$ that maximizes $P(i)$ and following the $C$ links back to the start of the path.

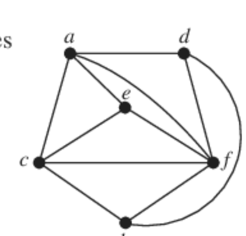
## Section 10.7

**1.** Yes **3.**



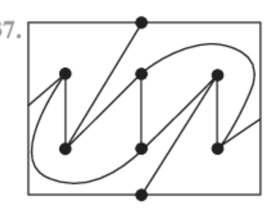

**5.** No

**7.** Yes



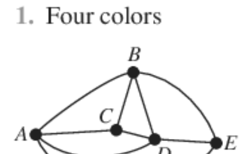

**9.** No **11.** A triangle is formed by the planar representation of the subgraph of $K_5$ consisting of the edges connecting $v_1$, $v_2$, and $v_3$. The vertex $v_4$ must be placed either within the triangle or outside of it. We will consider only the case when $v_4$ is inside the triangle; the other case is similar. Drawing the three edges from $v_1$, $v_2$, and $v_3$ to $v_4$ forms four regions. No matter which of these four regions $v_5$ is in, it is possible to join it to only three, and not all four, of the other vertices. **13.** 8 **15.** Because there are no loops or multiple edges and no simple circuits of length 3, and the degree of the unbounded region is at least 4, each region has degree at least 4. Thus $2e \geq 4r$, or $r \leq e/2$. But $r = e - v + 2$, so we have $e - v + 2 \leq e/2$, which implies that $e \leq 2v - 4$. **17.** As in the argument in the proof of Corollary 1, we have $2e \geq 5r$ and $r = e - v + 2$. Thus $e - v + 2 \leq 2e/5$, which implies that $e \leq (5/3)v - (10/3)$. **19.** Only (a) and (c) **21.** Not homeomorphic to $K_{3,3}$ **23.** Planar **25.** Nonplanar **27. a)** 1 **b)** 3 **c)** 9 **d)** 2 **e)** 4 **f)** 16 **29.** Draw $K_{m,n}$ as described in the hint. The number of crossings is four times the number in the first quadrant. The vertices on the $x$-axis to the right of the origin are $(1, 0), (2, 0), \ldots, (m/2, 0)$ and the vertices on the $y$-axis above the origin are $(0, 1), (0, 2), \ldots, (0, n/2)$. We obtain all crossings by choosing any two numbers $a$ and $b$ with $1 \leq a < b \leq m/2$ and two numbers $r$ and $s$ with $1 \leq r < s \leq n/2$; we get exactly one crossing in the graph between the edge connecting $(a, 0)$ and $(0, s)$ and the edge connecting $(b, 0)$ and $(0, r)$. Hence, the number of crossings in the first quadrant is $C\left(\frac{m}{2}, 2\right) \cdot C\left(\frac{n}{2}, 2\right) = \frac{(m/2)(m/2-1)}{2} \cdot \frac{(n/2)(n/2-1)}{2}$. Hence, the total number of crossings is $4 \cdot mn(m-2)(n-2)/64 = mn(m-2)(n-2)/16$. **31. a)** 2 **b)** 2 **c)** 2 **d)** 2 **e)** 2 **f)** 2 **33.** The formula is valid for $n \leq 4$. If $n > 4$, by Exercise 32 the thickness of $K_n$ is at least $C(n, 2)/(3n - 6) = (n + 1 + \frac{2}{n-2})/6$ rounded up. Because this quantity is never an integer, it equals $\lfloor (n+7)/6 \rfloor$. **35.** This follows from Exercise 34 because $K_{m,n}$ has $mn$ edges and $m + n$ vertices and has no triangles because it is bipartite.

**37.**

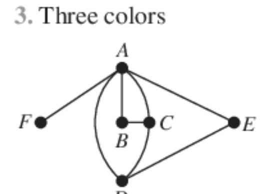


## Section 10.8

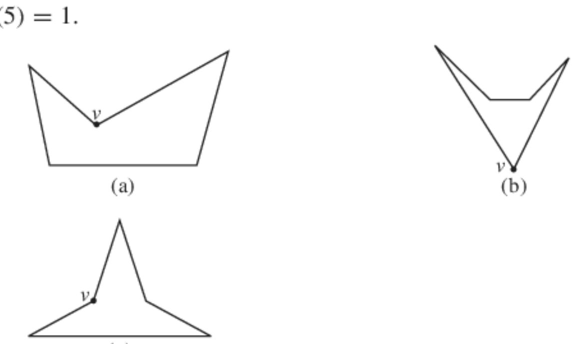
**1.** Four colors




**3.** Three colors




**5.** 3 **7.** 3 **9.** 2 **11.** 3 **13.** Graphs with no edges **15.** 3 if $n$ is even, 4 if $n$ is odd **17.** Period 1: Math 115, Math 185; period 2: Math 116, CS 473; period 3: Math 195, CS 101; period 4: CS 102; period 5: CS 273 **19.** 5 **21.** Exercise 5: 3 Exercise 6: 6 Exercise 7: 3 Exercise 8: 4 Exercise 9: 3 Exercise 10: 6 Exercise 11: 4 **23. a)** 2 if $n$ is even, 3 if $n$ is odd **b)** $n$ **25.** Two edges that have the same color share no endpoints. Therefore if more than $n/2$ edges were colored the same, the graph would have more than $2(n/2) = n$ vertices. **27.** 5 **29.** Color 1: *e, f, d*; color 2: *c, a, i, g*; color 3: *h, b, j* **31.** Color $C_6$ **33.** Four colors are needed to color $W_n$ when $n$ is an odd integer greater than 1, because three colors are needed for the rim (see Example 4), and the center vertex, being adjacent to all the rim vertices, will require a fourth color. To see that the graph obtained from $W_n$ by deleting one edge can be colored with three colors, consider two cases. If we remove a rim edge, then we can color the rim with two colors, by starting at an endpoint of the removed edge and using the colors alternately around the portion of the rim that remains. The third color is then assigned to the center vertex. If we remove a spoke edge, then we can color the rim by assigning color #1 to the rim endpoint of the removed edge and colors #2 and #3 alternately to the remaining vertices on the rim, and then assign color #1 to the center. **35.** Suppose that $G$ is chromatically $k$-critical but has a vertex $v$ of degree $k-2$ or less. Remove from $G$ one of the edges incident to $v$. By definition of "$k$-critical," the resulting graph can be colored with $k-1$ colors. Now restore the missing edge and use this coloring for all vertices except $v$. Because we had a proper coloring of the smaller graph, no two adjacent vertices have the same color. Furthermore, $v$ has at most $k-2$ neighbors, so we can color $v$ with an unused color to obtain a proper $(k-1)$-coloring of $G$. This contradicts the fact that $G$ has chromatic number $k$. Therefore, our assumption was wrong, and every vertex of $G$ must have degree at least $k-1$. **37. a)** 6 **b)** 7 **c)** 9 **d)** 11 **39.** Represent frequencies by colors and zones by vertices. Join two vertices with an edge if the zones these vertices represent interfere with one another. Then a $k$-tuple coloring is precisely an assignment of frequencies that avoids interference. **41.** We use induction on the number of vertices of the graph. Every graph with five or fewer vertices can be colored with five or fewer colors, because each vertex can get a different color. That takes care of the basis case(s). So we assume that all graphs with $k$ vertices can be 5-colored and consider a graph $G$ with $k+1$ vertices. By Corollary 2 in Section 10.7, $G$ has a vertex $v$ with degree at most 5. Remove $v$ to form the graph $G'$. Because $G'$ has only $k$ vertices, we 5-color it by the inductive hypothesis. If the neighbors of $v$ do not use all five colors, then we can 5-color $G$ by assigning to $v$ a color not used by any of its neighbors. The difficulty arises if $v$ has five neighbors, and each has a different color in the 5-coloring of $G'$. Suppose that the neighbors of $v$, when considered in clockwise order around $v$, are $a, b, c, m$, and $p$. (This order is determined by the clockwise order of the curves representing the edges incident to $v$.) Suppose that the colors of the neighbors are azure, blue, chartreuse, magenta, and purple, respectively. Consider the azure-chartreuse subgraph (i.e., the vertices in $G$ colored azure or chartreuse and all the edges between them). If $a$ and $c$ are not in the same component of this graph, then in the component containing $a$ we can interchange these two colors (make the azure vertices chartreuse and vice versa), and $G'$ will still be properly colored. That makes $a$ chartreuse, so we can now color $v$ azure, and $G$ has been properly colored. If $a$ and $c$ are in the same component, then there is a path of vertices alternately colored azure and chartreuse joining $a$ and $c$. This path together with edges $av$ and $vc$ divides the plane into two regions, with $b$ in one of them and $m$ in the other. If we now interchange blue and magenta on all the vertices in the same region as $b$, we will still have a proper coloring of $G'$, but now blue is available for $v$. In this case, too, we have found a proper coloring of $G$. This completes the inductive step, and the theorem is proved. **43.** We follow the hint. Because the measures of the interior angles of a pentagon total 540°, there cannot be as many as three interior angles of measure more than 180° (reflex angles). If there are no reflex angles, then the pentagon is convex, and a guard placed at any vertex can see all points. If there is one reflex angle, then the pentagon must look essentially like figure (a) below, and a guard at vertex $v$ can see all points. If there are two reflex angles, then they can be adjacent or nonadjacent (figures (b) and (c)); in either case, a guard at vertex $v$ can see all points. [In figure (c), choose the reflex vertex closer to the bottom side.] Thus for all pentagons, one guard suffices, so $g(5) = 1$.



(a) (b) (c)

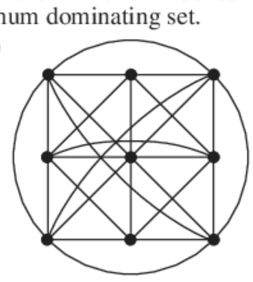
**45.** The figure suggested in the hint (generalized to have $k$ prongs for any $k \geq 1$) has $3k$ vertices. The sets of locations from which the tips of different prongs are visible are disjoint. Therefore, a separate guard is needed for each of the $k$ prongs, so at least $k$ guards are needed. This shows that $g(3k) \geq k = \lfloor 3k/3 \rfloor$. If $n = 3k + i$, where $0 \leq i \leq 2$, then $g(n) \geq g(3k) \geq k = \lfloor n/3 \rfloor$.

## Supplementary Exercises

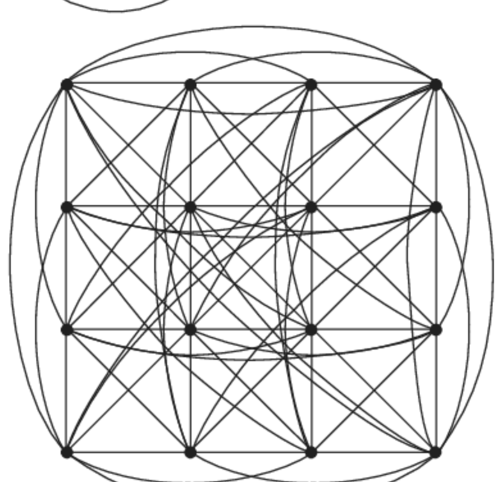
**1.** 2500 **3.** Yes **5.** Yes **7.** $\sum_{i=1}^{m} n_i$ vertices, $\sum_{i<j} n_i n_j$ edges **9. a)** If $x \in N(A \cup B)$, then $x$ is adjacent to some vertex $v \in A \cup B$. WOLOG suppose $v \in A$; then $x \in N(A)$ and therefore also in $N(A) \cup N(B)$. Conversely, if $x \in N(A) \cup N(B)$, then WOLOG suppose $x \in N(A)$. Thus $x$ is adjacent to some vertex $v \in A \subseteq A \cup B$, so $x \in N(A \cup B)$. **b)** If $x \in N(A \cap B)$, then $x$ is adjacent to some vertex $v \in A \cap B$. Since both $v \in A$ and $v \in B$, it follows that $x \in N(A)$ and $x \in N(B)$, whence $x \in N(A) \cap N(B)$. For the counterexample, let $G = (\{u, v, w\}, \{\{u, v\}, \{v, w\}\})$, $A = \{u\}$, and $B = \{w\}$. **11.** $(c, a, p, x, n, m)$ and many others **13.** $(c, d, a, b)$ and many others **15.** 6 times the

number of triangles divided by the number of paths of length 2 **17. a)** The probability that two actors each of whom has appeared in a film with a randomly chosen actor have appeared in a film together **b)** The probability that two of a randomly chosen person's Facebook friends are themselves Facebook friends **c)** The probability that two of a randomly chosen person's coauthors are themselves coauthors **d)** The probability that two proteins that each interact with a randomly chosen protein interact with each other **e)** The probability that two routers each of which has a communications link to a randomly chosen router are themselves linked **19.** Complete subgraphs containing the following sets of vertices: $\{b, c, e, f\}$, $\{a, b, g\}$, $\{a, d, g\}$, $\{d, e, g\}$, $\{b, e, g\}$ **21.** Complete subgraphs containing the following sets of vertices: $\{b, c, d, j, k\}$, $\{a, b, j, k\}$, $\{e, f, g, i\}$, $\{a, b, i\}$, $\{a, i, j\}$, $\{b, d, e\}$, $\{b, e, i\}$, $\{b, i, j\}$, $\{g, h, i\}$, $\{h, i, j\}$ **23.** $\{c, d\}$ is a minimum dominating set.

**25. a)**



**b)**



**27. a)** 1 **b)** 2 **c)** 3 **29. a)** A path from $u$ to $v$ in a graph $G$ induces a path from $f(u)$ to $f(v)$ in an isomorphic graph $H$. **b)** Suppose $f$ is an isomorphism from $G$ to $H$. If $v_0, v_1, \ldots, v_n, v_0$ is a Hamilton circuit in $G$, then $f(v_0), f(v_1), \ldots, f(v_n), f(v_0)$ must be a Hamilton circuit in $H$ because it is still a circuit and $f(v_i) \neq f(v_j)$ for $0 \leq i < j \leq n$. **c)** Suppose $f$ is an isomorphism from $G$ to $H$. If $v_0, v_1, \ldots, v_n, v_0$ is an Euler circuit in $G$, then $f(v_0), f(v_1), \ldots, f(v_n), f(v_0)$ must be an Euler circuit in $H$ because it is a circuit that contains each edge exactly once. **d)** Two isomorphic graphs must have the same crossing number because they can be drawn exactly the same way in the plane. **e)** Suppose $f$ is an isomorphism from $G$ to $H$. Then $v$ is isolated in $G$ if and only if $f(v)$ is isolated in $H$. Hence, the graphs must have the same number of isolated vertices. **f)** Suppose $f$ is an isomorphism from $G$ to $H$. If $G$ is bipartite, then the vertex set of $G$ can be partitioned into $V_1$ and $V_2$ with no edge connecting vertices within $V_1$ or vertices within $V_2$. Then the vertex set of $H$ can be partitioned into $f(V_1)$ and $f(V_2)$ with no edge connecting vertices within $f(V_1)$ or vertices within $f(V_2)$. **31.** 3 **33. a)** Yes **b)** No **35.** No **37.** Yes **39.** If $e$ is a cut edge with endpoints $u$ and $v$, then if we direct $e$ from $u$ to $v$, there will be no path in the directed graph from $v$ to $u$, or else $e$ would not have been a cut edge. Similar reasoning works if we direct $e$ from $v$ to $u$. **41.** $n - 1$ **43.** Let the vertices represent the chickens. We include the edge $(u, v)$ in the graph if and only if chicken $u$ dominates chicken $v$. **45.** By the handshaking theorem, the average vertex degree is $2m/n$, which equals the minimum degree; it follows that all the vertex degrees are equal. **47.** $K_{3,3}$ and the skeleton of a triangular prism **49. a)** A Hamilton circuit in the graph exactly corresponds to a seating of the knights at the Round Table such that adjacent knights are friends. **b)** The degree of each vertex in this graph is at least $2n - 1 - (n - 1) = n \geq (2n/2)$, so by Dirac's theorem, this graph has a Hamilton circuit. **c)** $a, b, d, f, g, z$ **51. a)** 4 **b)** 2 **c)** 3 **d)** 4 **e)** 4 **f)** 2 **53. a)** Suppose that $G = (V, E)$. Let $a, b \in V$. We must show that the distance between $a$ and $b$ in $\overline{G}$ is at most 2. If $\{a, b\} \notin E$ this distance is 1, so assume $\{a, b\} \in E$. Because the diameter of $G$ is greater than 3, there are vertices $u$ and $v$ such that the distance in $G$ between $u$ and $v$ is greater than 3. Either $u$ or $v$, or both, is not in the set $\{a, b\}$. Assume that $u$ is different from both $a$ and $b$. Either $\{a, u\}$ or $\{b, u\}$ belongs to $E$; otherwise $a, u, b$ would be a path in $\overline{G}$ of length 2. So, without loss of generality, assume $\{a, u\} \in E$. Thus $v$ cannot be $a$ or $b$, and by the same reasoning either $\{a, v\} \in E$ or $\{b, v\} \in E$. In either case, this gives a path of length less than or equal to 3 from $u$ to $v$ in $G$, a contradiction. **b)** Suppose $G = (V, E)$. Let $a, b \in V$. We must show that the distance between $a$ and $b$ in $\overline{G}$ does not exceed 3. If $\{a, b\} \notin E$, the result follows, so assume that $\{a, b\} \in E$. Because the diameter of $G$ is greater than or equal to 3, there exist vertices $u$ and $v$ such that the distance in $G$ between $u$ and $v$ is greater than or equal to 3. Either $u$ or $v$, or both, is not in the set $\{a, b\}$. Assume $u$ is different from both $a$ and $b$. Either $\{a, u\} \in E$ or $\{b, u\} \in E$; otherwise $a, u, b$ is a path of length 2 in $\overline{G}$. So, without loss of generality, assume $\{a, u\} \in E$. Thus $v$ is different from $a$ and from $b$. If $\{a, v\} \in E$, then $u, a, v$ is a path of length 2 in $G$, so $\{a, v\} \notin E$ and thus $\{b, v\} \in E$ (or else there would be a path $a, v, b$ of length 2 in $\overline{G}$). Hence, $\{u, b\} \notin E$; otherwise $u, b, v$ is a path of length 2 in $G$. Thus, $a, v, u, b$ is a path of length 3 in $\overline{G}$, as desired. **55.** $a, b, e, z$ **57.** $a, c, d, f, g, z$ **59.** If $G$ is planar, then because $e \leq 3v - 6$, $G$ has at most 27 edges. (If $G$ is not connected it has even fewer edges.) Similarly, $\overline{G}$ has at most 27 edges. But the union of $G$ and $\overline{G}$ is $K_{11}$, which has 55 edges, and $55 > 27 + 27$. **61.** Suppose that $G$ is colored with $k$ colors and has independence number $i$. Because each color class must be an independent set, each color class has no more than $i$ elements. Thus there are at most $ki$

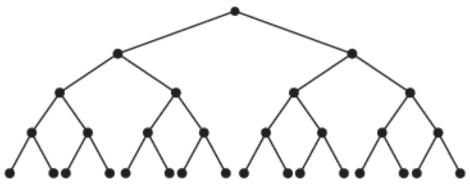
vertices. 63. **a)** $C(n, m)p^m(1-p)^{n-m}$ **b)** $np$ **c)** To generate a labeled graph $G$, as we apply the process to pairs of vertices, the random number $x$ chosen must be less than or equal to $1/2$ when $G$ has an edge between that pair of vertices and greater than $1/2$ when $G$ has no edge there. Hence, the probability of making the correct choice is $1/2$ for each edge and $1/2^{C(n,2)}$ overall. Hence, all labeled graphs are equally likely. 65. Suppose $P$ is monotone increasing. If the property of not having $P$ were not retained whenever edges are removed from a simple graph, there would be a simple graph $G$ not having $P$ and another simple graph $G'$ with the same vertices but with some of the edges of $G$ missing that has $P$. But $P$ is monotone increasing, so because $G'$ has $P$, so does $G$ obtained by adding edges to $G'$. This is a contradiction. The proof of the converse is similar.

# CHAPTER 11

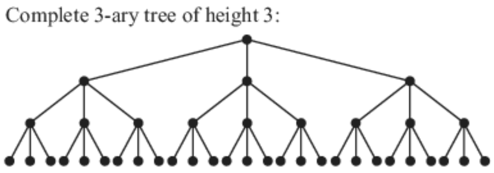
## Section 11.1

1. (a), (c), (e) 3. **a)** $a$ **b)** $a, b, c, d, f, h, j, q, t$ **c)** $e, g, i, k, l, m, n, o, p, r, s, u$ **d)** $q, r$ **e)** $c$ **f)** $p$ **g)** $f, b, a$ **h)** $e, f, l, m, n$ 5. No 7. Level 0: $a$; level 1: $b$, $c$, $d$; level 2: $e$ through $k$ (in alphabetical order); level 3: $l$ through $r$; level 4: $s$, $t$; level 5: $u$ 9. **a)** The entire tree **b)** $c$, $g$, $h$, $o$, $p$ and the four edges $cg$, $ch$, $ho$, $hp$ **c)** $e$ alone 11. **a)** 1 **b)** 2 13. **a)** 3 **b)** 9 15. **a)** The "only if" part is Theorem 2 and the definition of a tree. Suppose $G$ is a connected simple graph with $n$ vertices and $n-1$ edges. If $G$ is not a tree, it contains, by Exercise 14, an edge whose removal produces a graph $G'$, which is still connected. If $G'$ is not a tree, remove an edge to produce a connected graph $G''$. Repeat this procedure until the result is a tree. This requires at most $n-1$ steps because there are only $n-1$ edges. By Theorem 2, the resulting graph has $n-1$ edges because it has $n$ vertices. It follows that no edges were deleted, so $G$ was already a tree. **b)** Suppose that $G$ is a tree. By part (a), $G$ has $n-1$ edges, and by definition, $G$ has no simple circuits. Conversely, suppose that $G$ has no simple circuits and has $n-1$ edges. Let $c$ equal the number of components of $G$, each of which is necessarily a tree, say with $n_i$ vertices, where $\sum_{i=1}^{c} n_i = n$. By part (a), the total number of edges in $G$ is $\sum_{i=1}^{c}(n_i - 1) = n - c$. Since we are given that this equals $n-1$, it follows that $c = 1$, i.e., $G$ is connected and therefore satisfies the definition of a tree. 17. 9999 19. 2000 21. 999 23. 1,000,000 dollars 25. No such tree exists by Theorem 4 because it is impossible for $m = 2$ or $m = 84$.

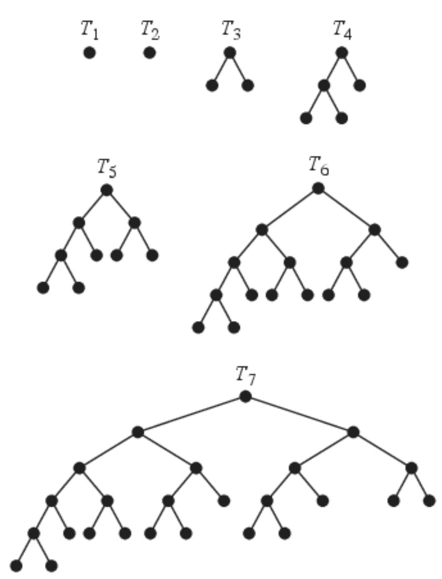
27. Complete binary tree of height 4:



Complete 3-ary tree of height 3:

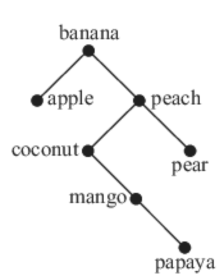


29. **a)** By Theorem 3 it follows that $n = mi + 1$. Because $i + l = n$, we have $l = n - i$, so $l = (mi + 1) - i = (m-1)i + 1$. **b)** We have $n = mi + 1$ and $i + l = n$. Hence, $i = n - l$. It follows that $n = m(n-l) + 1$. Solving for $n$ gives $n = (ml - 1)/(m-1)$. From $i = n - l$ we obtain $i = [(ml-1)/(m-1)] - l = (l-1)/(m-1)$. 31. $n - t$ 33. **a)** 1 **b)** 3 **c)** 5 35. **a)** The parent directory **b)** A subdirectory or contained file **c)** A subdirectory or contained file in the same parent directory **d)** All directories in the path name **e)** All subdirectories and files continued in the directory or a subdirectory of this directory, and so on **f)** The length of the path to this directory or file **g)** The depth of the system, i.e., the length of the longest path 37. Let $n = 2^k$, where $k$ is a positive integer. If $k = 1$, there is nothing to prove because we can add two numbers with $n - 1 = 1$ processor in $\log 2 = 1$ step. Assume we can add $n = 2^k$ numbers in $\log n$ steps using a tree-connected network of $n-1$ processors. Let $x_1, x_2, \ldots, x_{2n}$ be $2n = 2^{k+1}$ numbers that we wish to add. The tree-connected network of $2n-1$ processors consists of the tree-connected network of $n-1$ processors together with two new processors as children of each leaf. In one step we can use the leaves of the larger network to find $x_1 + x_2, x_3 + x_4, \ldots, x_{2n-1} + x_{2n}$, giving us $n$ numbers, which, by the inductive hypothesis, we can add in $\log n$ steps using the rest of the network. Because we have used $\log n + 1$ steps and $\log(2n) = \log 2 + \log n = 1 + \log n$, this completes the proof. 39. $c$ only 41. $c$ and $h$ 43. Suppose a tree $T$ has at least two centers. Let $u$ and $v$ be distinct centers, both with eccentricity $e$, with $u$ and $v$ not adjacent. Because $T$ is connected, there is a simple path $P$ from $u$ to $v$. Let $c$ be any other vertex on this path. Because the eccentricity of $c$ is at least $e$, there is a vertex $w$ such that the unique simple path from $c$ to $w$ has length at least $e$. Clearly, this path cannot contain both $u$ and $v$ or else there would be a simple circuit. In fact, this path from $c$ to $w$ leaves $P$ and does not return to $P$ once it, possibly, follows part of $P$ toward either $u$ or $v$. Without loss of generality, assume this path does not follow $P$ toward $u$. Then the path from $u$ to $c$ to $w$ is simple and of length more than $e$, a contradiction. Hence, $u$ and $v$ are adjacent. Now because any two centers are adjacent, if there were more than two centers, $T$ would contain $K_3$, a simple circuit, as a subgraph, which is a contradiction.

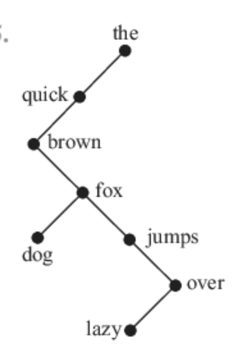
45. 

47. The statement is that *every* tree with $n$ vertices has a path of length $n-1$, and it was shown only that there exists a tree with $n$ vertices having a path of length $n-1$.

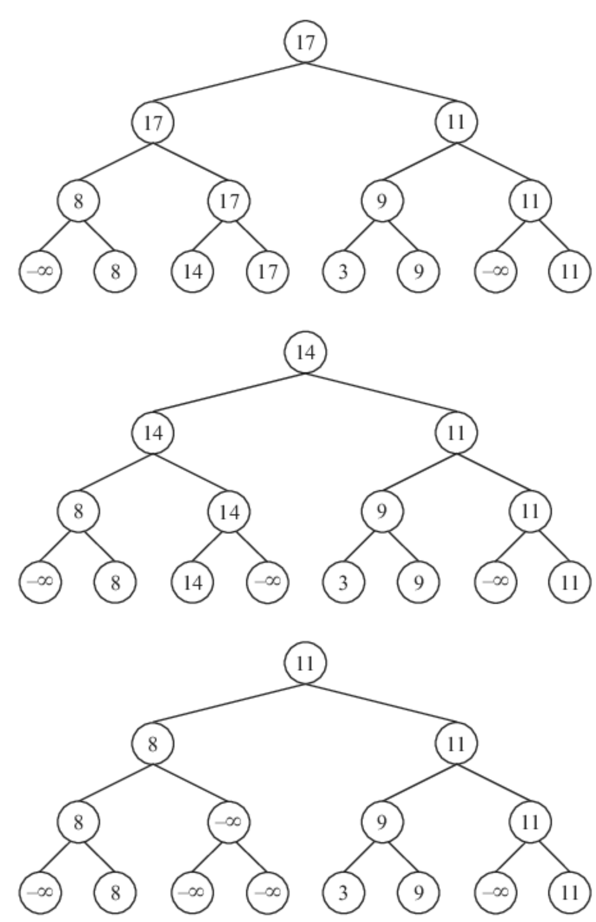
## Section 11.2

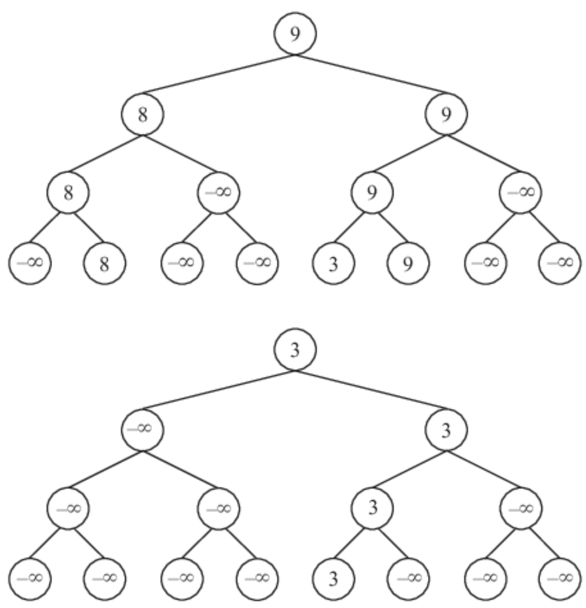
1. 

3. **a)** 3 **b)** 1 **c)** 4 **d)** 5

5. 

7. At least $\lceil \log_3 4 \rceil = 2$ weighings are needed, because there are only four outcomes (because it is not required to determine whether the coin is lighter or heavier). In fact, two weighings suffice. Begin by weighing coin 1 against coin 2. If they balance, weigh coin 1 against coin 3. If coin 1 and coin 3 are the same weight, coin 4 is the counterfeit coin, and if they are not the same weight, then coin 3 is the counterfeit coin. If coin 1 and coin 2 are not the same weight, again weigh coin 1 against coin 3. If they balance, coin 2 is the counterfeit coin; if they do not balance, coin 1 is the counterfeit coin. 9. At least $\lceil \log_3 13 \rceil = 3$ weighings are needed. In fact, three weighings suffice. Start by putting coins 1, 2, and 3 on the left-hand side of the balance and coins 4, 5, and 6 on the right-hand side. If equal, apply Example 3 to coins 1, 2, 7, 8, 9, 10, 11, and 12. If unequal, apply Example 3 to 1, 2, 3, 4, 5, 6, 7, and 8. 11. The least number is five. Call the elements $a, b, c$, and $d$. First compare $a$ and $b$; then compare $c$ and $d$. Without loss of generality, assume that $a < b$ and $c < d$. Next compare $a$ and $c$. Whichever is smaller is the smallest element of the set. Again without loss of generality, suppose $a < c$. Finally, compare $b$ with both $c$ and $d$ to completely determine the ordering. 13. The first two steps are shown in the text. After 22 has been identified as the second largest element, we replace the leaf 22 by $-\infty$ in the tree and recalculate the winner in the path from the leaf where 22 used to be up to the root. Next, we see that 17 is the third largest element, so we repeat the process: replace the leaf 17 by $-\infty$ and recalculate. Next, we see that 14 is the fourth largest element, so we repeat the process: replace the leaf 14 by $-\infty$ and recalculate. Next, we see that 11 is the fifth largest element, so we repeat the process: replace the leaf 11 by $-\infty$ and recalculate. The process continues in this manner. We determine that 9 is the sixth largest element, 8 is the seventh largest element, and 3 is the eighth largest element. The trees produced in all steps, except the second to last, are shown here.

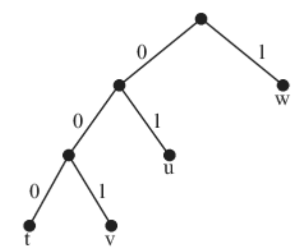




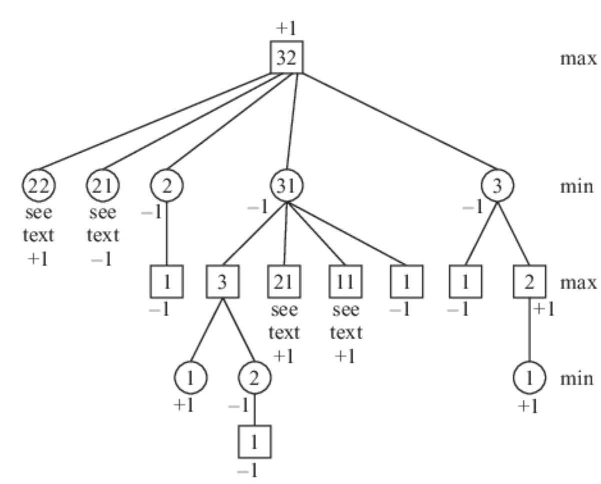
**15.** The value of a vertex is the list element currently there, and the label is the name (i.e., location) of the leaf responsible for that value.

**procedure** *tournament sort*$(a_1, \ldots, a_n)$
$k := \lceil \log n \rceil$
build a binary tree of height $k$
**for** $i := 1$ **to** $n$
  set the value of the $i$th leaf to be $a_i$ and its label to be itself
**for** $i := n + 1$ **to** $2^k$
  set the value of the $i$th leaf to be $-\infty$ and its label to be itself
**for** $i := k - 1$ **downto** 0
  **for** each vertex $v$ at level $i$
    set the value of $v$ to the larger of the values of its children and its label to be the label of the child with the larger value
**for** $i := 1$ **to** $n$
  $c_i :=$ value at the root
  let $v$ be the label of the root
  set the value of $v$ to be $-\infty$
  **while** the label at the root is still $v$
    $v := parent(v)$
    set the value of $v$ to the larger of the values of its children and its label to be the label of the child with the larger value
$\{c_1, \ldots, c_n$ is the list in nonincreasing order$\}$

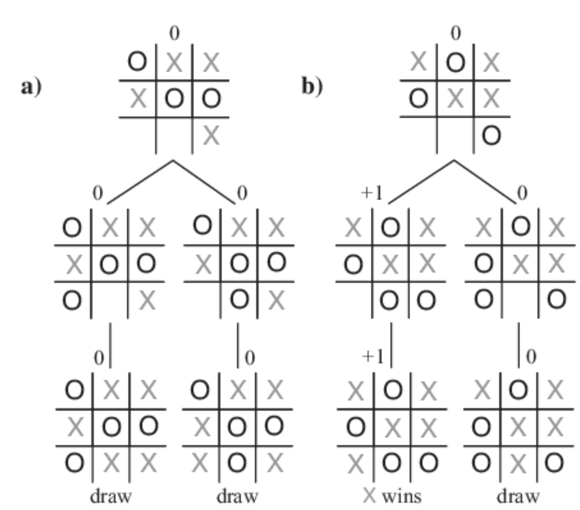
**17.** $k - 1$, where $n = 2^k$ **19. a)** Yes **b)** No **c)** Yes **d)** Yes **21.** $a$: 000, $e$: 001, $i$: 01, $k$: 1100, $o$: 1101, $p$: 11110, $u$: 11111 **23.** a: 11; b: 101; c: 100; d: 01; e: 00; 2.25 bits (Note: This coding depends on how ties are broken, but the average number of bits is always the same.) **25.** There are four possible answers in all, the one shown here and three more obtained from this one by swapping t and v and/or swapping u and w.



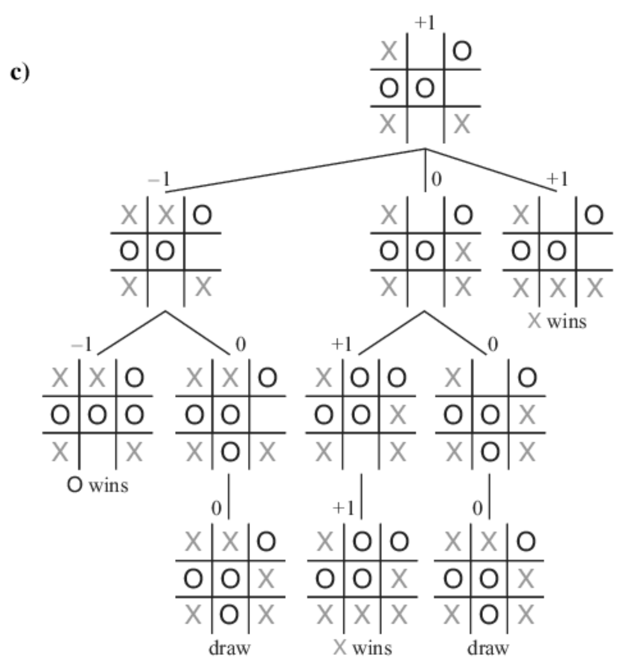
**27.** A:0001; B:101001; C:11001; D:00000; E:100; F:001100; G:001101; H:0101; I:0100; J:110100101; K:1101000; L:00001; M:10101; N:0110; O:0010; P:101000; Q:1101001000; R:1011; S:0111; T:111; U:00111; V:110101; W:11000; X:11010011; Y:11011; Z:1101001001 **29.** A:2; E:1; N:010; R:011; T:02; Z:00 **31.** $n$ **33.** Because the tree is rather large, we have indicated in some places to "see text." Refer to Figure 9; the subtree rooted at these square or circle vertices is exactly the same as the corresponding subtree in Figure 9. First player wins.

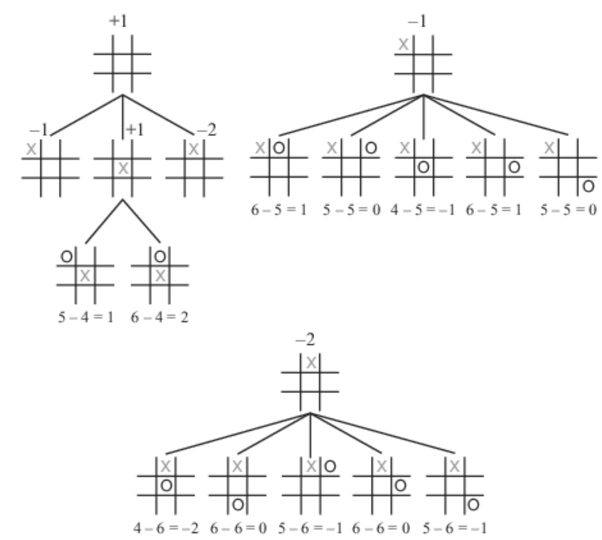



**35. a)** \$1 **b)** \$3 **c)** −\$3 **37.** See the figures shown next. **a)** 0 **b)** 0 **c)** 1 **d)** This position cannot have occurred in a game; this picture is impossible.



**c)**

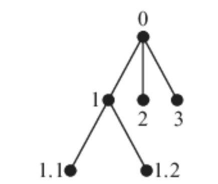



**39.** Proof by strong induction: *Basis step:* When there are $n = 2$ stones in each pile, if first player takes two stones from a pile, then second player takes one stone from the remaining pile and wins. If first player takes one stone from a pile, then second player takes two stones from the other pile and wins. *Inductive step:* Assume inductive hypothesis that second player can always win if the game starts with two piles of $j$ stones for all $2 \leq j \leq k$, where $k \geq 2$, and consider a game with two piles containing $k + 1$ stones each. If first player takes all the stones from one of the piles, then second player takes all but one stone from the remaining pile and wins. If first player takes all but one stone from one of the piles, then second player takes all the stones from the other pile and wins. Otherwise first player leaves $j$ stones in one pile, where $2 \leq j \leq k$, and $k + 1$ stones in the other pile. Second player takes the same number of stones from the larger pile, also leaving $j$ stones there. At this point the game consists of two piles of $j$ stones each. By the inductive hypothesis, the second player in that game, who is also the second player in our actual game, can win, and the proof by strong induction is complete. **41.** 7; 49 **43.** Value of tree is 1. Note: The second and third trees are the subtrees of the two children of the root in the first tree whose subtrees are not shown because of space limitations. They should be thought of as spliced into the first picture.

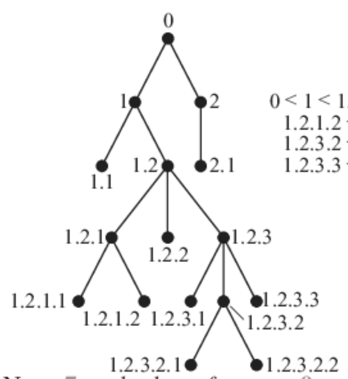
## Section 11.3

**1.**




0 < 1 < 1.1 < 1.2 < 2 < 3

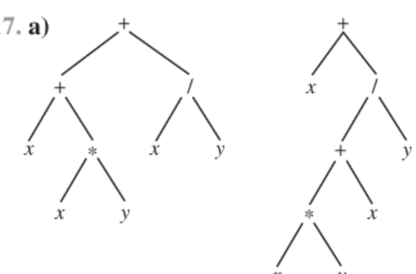
**3.**




0 < 1 < 1.1 < 1.2 < 1.2.1 < 1.2.1.1 < 1.2.1.2 < 1.2.2 < 1.2.3 < 1.2.3.1 < 1.2.3.2 < 1.2.3.2.1 < 1.2.3.2.2 < 1.2.3.3 < 2 < 2.1

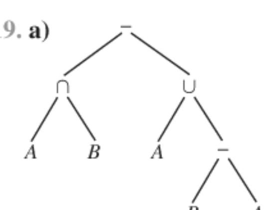
**5.** No **7.** $a, b, d, e, f, g, c$ **9.** $a, b, e, k, l, m, f, g, n, r, s, c, d, h, o, i, j, p, q$ **11.** $d, b, i, e, m, j, n, o, a, f, c, g, k, h, p, l$ **13.** $d, f, g, e, b, c, a$ **15.** $k, l, m, e, f, r, s, n, g, b, c, o, h, i, p, q, j, d, a$

**17. a)**



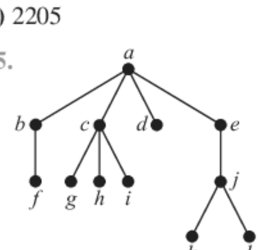

**b)** $++x*xy/xy, +x/+*xyxy$ **c)** $xy*+xy/+, xxy*x+y/+$
**d)** $((x + (x * y)) + (x/y)), (x + (((x * y) + x)/y))$

**19. a)**




**b)** $- \cap A\ B \cup A - B\ A$ **c)** $A\ B \cap A\ B\ A - \cup -$
**d)** $((A \cap B) - (A \cup (B - A)))$ **21.** 14 **23. a)** 1 **b)** 1 **c)** 4
**d)** 2205

**25.**

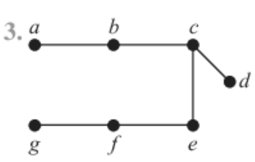



**27.** Use mathematical induction. The result is trivial for a list with one element. Assume the result is true for a list with $n$ elements. For the inductive step, start at the end. Find the sequence of vertices at the end of the list starting with the last leaf, ending with the root, each vertex being the last child of the one following it. Remove this leaf and apply the inductive hypothesis. **29.** $c, d, b, f, g, h, e, a$ in each case **31.** Proof by mathematical induction. Let $S(X)$ and $O(X)$ represent the number of symbols and number of operators in the well-formed formula $X$, respectively. The statement is true for well-formed formulae of length 1, because they have 1 symbol and 0 operators. Assume the statement is true for all well-formed formulae of length less than $n$. A well-formed formula of length $n$ must be of the form $*XY$, where $*$ is an operator and $X$ and $Y$ are well-formed formulae of length less than $n$. Then by the inductive hypothesis $S(*XY) = S(X) + S(Y) = [O(X) + 1] + [O(Y) + 1] = O(X) + O(Y) + 2$. Because $O(*XY) = 1 + O(X) + O(Y)$, it follows that $S(*XY) = O(*XY) + 1$. **33.** $x\ y + z\ x \circ + x \circ$, $xyz + + yx++$, $xyxy\circ\circ xy\circ\circ z\circ+$, $xz\times$, $zz+\circ$, $yyyy\circ\circ\circ$, $zx+yz+\circ$, for instance

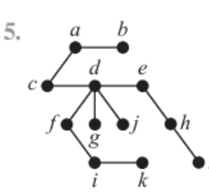
## Section 11.4

**1.** $m - n + 1$

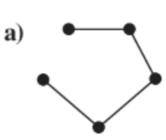
**3.**




**5.**



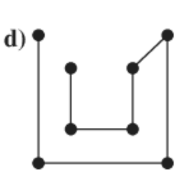

**7. a)**



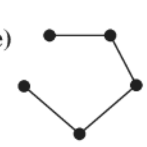
**b)**

**c)**

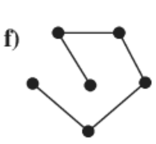
**d)**

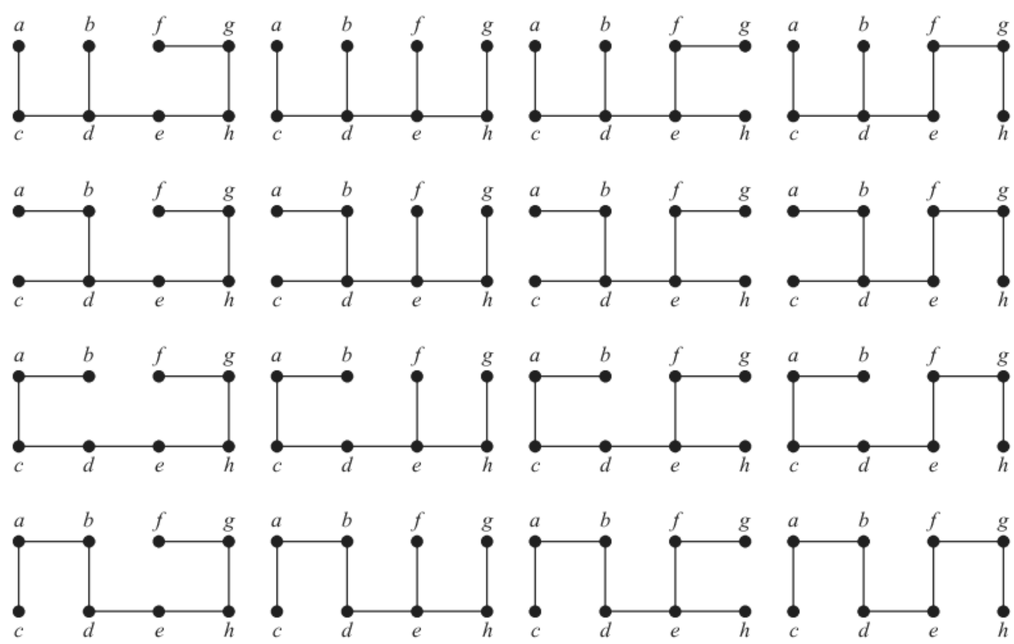


**e)**

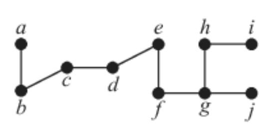


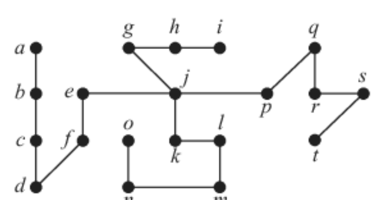
**f)**



9. 

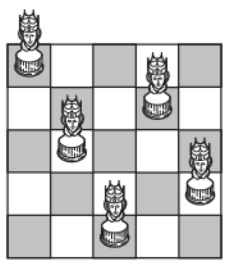
11. **a)** 3 **b)** 16 **c)** 4 **d)** 5

13. 

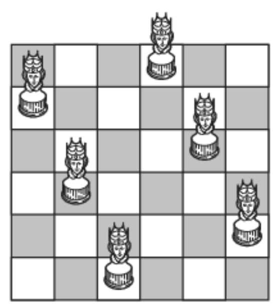

15. 


17. **a)** A path of length 6 **b)** A path of length 5 **c)** A path of length 6 **d)** Depends on order chosen to visit the vertices; may be a path of length 7 19. With breadth-first search, the initial vertex is the middle vertex, and the $n$ spokes are added to the tree as this vertex is processed. Thus, the resulting tree is $K_{1,n}$. With depth-first search, we start at the vertex in the middle of the wheel and visit a neighbor—one of the vertices on the rim. From there we move to an adjacent vertex on the rim, and so on all the way around until we have reached every vertex. Thus, the resulting spanning tree is a path of length $n$. 21. With breadth-first search, we fan out from a vertex of degree $m$ to all the vertices of degree $n$ as the first step. Next, a vertex of degree $n$ is processed, and the edges from it to all the remaining vertices of degree $m$ are added. The result is a $K_{1,n-1}$ and a $K_{1,m-1}$ with their centers joined by an edge. With depth-first search, we travel back and forth from one partite set to the other until we can go no further. If $m = n$ or $m = n - 1$, then we get a path of length $m + n - 1$. Otherwise, the path ends while some vertices in the larger partite set have not been visited, so we back up one link in the path to a vertex $v$ and then successively visit the remaining vertices in that set from $v$. The result is a path with extra pendant edges coming out of one end of the path. 23. A possible set of flights to discontinue are: Boston–New York, Detroit–Boston, Boston–Washington, New York–Washington, New York–Chicago, Atlanta–Washington, Atlanta–Dallas, Atlanta–Los Angeles, Atlanta–St. Louis, St. Louis–Dallas, St. Louis–Detroit, St. Louis–Denver, Dallas–San Diego, Dallas–Los Angeles, Dallas–San Francisco, San Diego–Los Angeles, Los Angeles–San Francisco, San Francisco–Seattle. 25. Proof by induction on the length of the path: If the path has length 0, then the result is trivial. If the length is 1, then $u$ is adjacent to $v$, so $u$ is at level 1 in the breadth-first spanning tree. Assume that the result is true for paths of length $l$. If the length of a path is $l + 1$, let $u'$ be the next-to-last vertex in a shortest path from $v$ to $u$. By the inductive hypothesis, $u'$ is at level $l$ in the breadth-first spanning tree. If $u$ were at a level not exceeding $l$, then clearly the length of the shortest path from $v$ to $u$ would also not exceed $l$. So $u$ has not been added to the breadth-first spanning tree yet after the vertices of level $l$ have been added. Because $u$ is adjacent to $u'$, it will be added at level $l + 1$ (although the edge connecting $u'$ and $u$ is not necessarily added). 27. **a)** No solution

**b)**

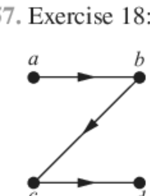


**c)**

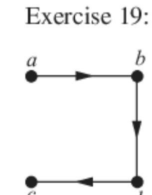


29. Start at a vertex and proceed along a path without repeating vertices as long as possible, allowing the return to the start after all vertices have been visited. When it is impossible to continue along a path, backtrack and try another extension of the current path. 31. Take the union of the spanning trees of the connected components of $G$. They are disjoint, so the result is a forest. 33. $m - n + c$ 35. Assume that we wish to find the length of a shortest path from $v_1$ to every other vertex of $G$ using Algorithm 1. In line 2 of that algorithm, add $L(v_1) := 0$, and add the following as a third step in the **then** clause at the end: $L(w) := 1 + L(v)$. 37. Add an instruction to the BFS algorithm to mark each vertex as it is encountered. When BFS terminates we have found (all the vertices of) one component of the graph. Repeat, starting at an unmarked vertex, and continue in this way until all vertices have been marked. 39. Trees 41. Use depth-first search on each component. 43. If an edge $uv$ is not followed while we are processing vertex $u$ during the depth-first search process, then it must be the case that the vertex $v$ had already been visited. There are two cases. If vertex $v$ was visited after we started processing $u$, then, because we are not finished processing $u$ yet, $v$ must appear in the subtree rooted at $u$ (and hence, must be a descendant of $u$). On the other hand, if the processing of $v$ had already begun before we started processing $u$, then why wasn't this edge followed at that time? It must be that we had not finished processing $v$, in other words, that we are still forming the subtree rooted at $v$, so $u$ is a descendant of $v$, and hence, $v$ is an ancestor of $u$. 45. Certainly these two procedures produce the identical spanning trees if the graph we are working with is a tree itself, because in this case there is only one spanning tree (the whole graph). This is the only case in which that happens, however. If the original graph has any other edges, then by Exercise 43 they must be back edges and hence, join a vertex to an ancestor or descendant, whereas by Exercise 34, they must connect vertices at the same level or at levels that differ by 1. Clearly these two possibilities are mutually exclusive. Therefore there can be no edges other than tree edges if the two spanning trees are to be the same. 47. Because the edges not in the spanning tree are not followed in the process, we can ignore them. Thus we can assume that the graph was a rooted tree to begin with. The basis step is trivial (there is only one vertex), so we assume the inductive hypothesis that breadth-first search applied to trees with $n$ vertices have their vertices visited in order of their level in the tree and consider a tree $T$ with $n + 1$ vertices. The last vertex to be visited during breadth-first search of this tree, say $v$, is the one that was added last to the list of vertices waiting to be processed. It was added when its parent, say $u$, was being processed. We must show that $v$ is at the lowest (bottom-most, i.e., numerically greatest) level of the tree. Suppose not; say vertex $x$, whose parent is vertex $w$, is at a lower level. Then $w$ is at a lower level than $u$. Clearly $v$ must be a leaf, because any child of $v$ could not have been seen before $v$ is seen. Consider the tree $T'$ obtained from $T$ by deleting $v$. By the inductive hypothesis, the vertices in $T'$ must be processed in order of their level in $T'$ (which is the same as their level in $T$, and the absence of $v$ in $T'$ has no effect on the rest of the algorithm). Therefore $u$ must have been processed before $w$, and therefore $v$ would have joined the waiting list before $x$ did, a contradiction. Therefore $v$ is at the bottom-most level of the tree, and the proof is complete. 49. We modify the pseudocode given in Algorithm 2 by initializing $m$ to be 0 at the beginning of the algorithm, and adding the statements "$m := m + 1$" and "assign $m$ to vertex $v$" after the statement that removes vertex $v$ from $L$. 51. If a directed edge $uv$ is not followed while we are processing its tail $u$ during the depth-first search process, then it must be the case that its head $v$ had already been visited. There are three cases. If vertex $v$ was visited after we started processing $u$, then, because we are not finished processing $u$ yet, $v$ must appear in the subtree rooted at $u$ (and hence, must be a descendant of $u$), so we have a forward edge. Otherwise, the processing of $v$ must have already begun before we started processing $u$. If it had not yet finished (i.e., we are still forming the subtree rooted at $v$), then $u$ is a descendant of $v$, and hence, $v$ is an ancestor of $u$ (we have a back edge). Finally, if the processing of $v$ had already finished, then by definition we have a cross edge. 53. Let $T$ be the spanning tree constructed in Figure 3 and $T_1$, $T_2$, $T_3$, and $T_4$ the spanning trees in Figure 4. Denote by $d(T', T'')$ the distance between $T'$ and $T''$. Then $d(T, T_1) = 6$, $d(T, T_2) = 4$, $d(T, T_3) = 4$, $d(T, T_4) = 2$, $d(T_1, T_2) = 4$, $d(T_1, T_3) = 4$, $d(T_1, T_4) = 6$, $d(T_2, T_3) = 4$, $d(T_2, T_4) = 2$, and $d(T_3, T_4) = 4$. 55. Suppose $e_1 = \{u, v\}$ is as specified. Then $T_2 \cup \{e_1\}$ contains a simple circuit $C$ containing $e_1$. The graph $T_1 - \{e_1\}$ has two connected components; the endpoints of $e_1$ are in different components. Travel $C$ from $u$ in the direction opposite to $e_1$ until you come to the first vertex in the same component as $v$. The edge just crossed is $e_2$. Clearly, $T_2 \cup \{e_1\} - \{e_2\}$ is a tree, because $e_2$ was on $C$. Also $T_1 - \{e_1\} \cup \{e_2\}$ is a tree, because $e_2$ reunited the two components.

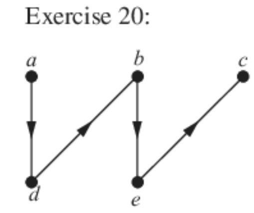
57. Exercise 18:



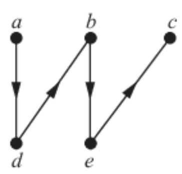

Exercise 19:



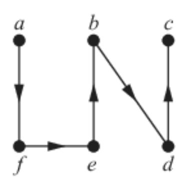

Exercise 20:



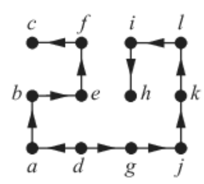
Exercise 21:



Exercise 22:



Exercise 23:

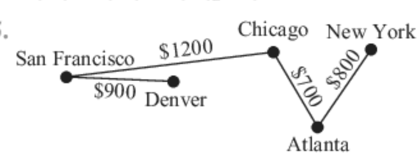


59. First construct an Euler circuit in the directed graph. Then delete from this circuit every edge that goes to a vertex previously visited. 61. According to Exercise 60, a directed graph contains a circuit if and only if there are any back edges. We can detect back edges as follows. Add a marker on each vertex $v$ to indicate what its status is: not yet seen (the initial situation), seen (i.e., put into $T$) but not yet finished (i.e., $visit(v)$ has not yet terminated), or finished (i.e., $visit(v)$ has terminated). A few extra lines in Algorithm 1 will accomplish this bookkeeping. Then to determine whether a directed graph has a circuit, we just have to check when looking at edge $uv$ whether the status of $v$ is "seen." If that ever happens, then we know there is a circuit; if not, then there is no circuit.

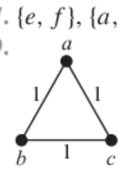
## Section 11.5

1. Deep Springs–Oasis, Oasis–Dyer, Oasis–Silver Peak, Silver Peak–Goldfield, Lida–Gold Point, Gold Point–Beatty, Lida–Goldfield, Goldfield–Tonopah, Tonopah–Manhattan, Tonopah–Warm Springs 3. $\{e, f\}$, $\{c, f\}$, $\{e, h\}$, $\{h, i\}$, $\{b, c\}$, $\{b, d\}$, $\{a, d\}$, $\{g, h\}$

5.

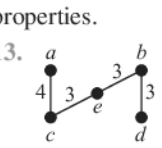



7. $\{e, f\}$, $\{a, d\}$, $\{h, i\}$, $\{b, d\}$, $\{c, f\}$, $\{e, h\}$, $\{b, c\}$, $\{g, h\}$
9.

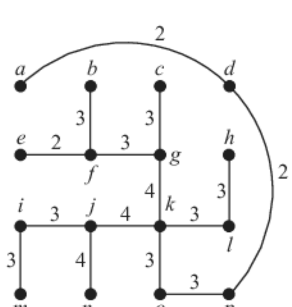


11. Instead of choosing minimum-weight edges at each stage, choose maximum-weight edges at each stage with the same properties.

13.

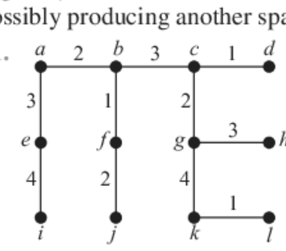


15.



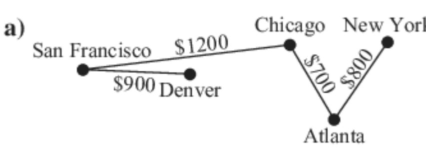
17. First find a minimum spanning tree $T$ of the graph $G$ with $n$ edges. Then for $i = 1$ to $n - 1$, delete only the $i$th edge of $T$ from $G$ and find a minimum spanning tree of the remaining graph. Pick the one of these $n - 1$ trees with the shortest length. 19. If all edges have different weights, then a contradiction is obtained in the proof that Prim's algorithm works when an edge $e_{k+1}$ is added to $T$ and an edge $e$ is deleted, instead of possibly producing another spanning tree.

21.

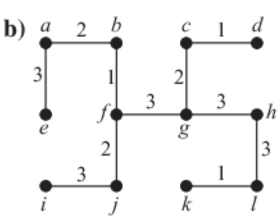


23. Same as Kruskal's algorithm, except start with $T :=$ this set of edges and iterate from $i = 1$ to $i = n - 1 - s$, where $s$ is the number of edges you start with.

25. **a)**




**b)**



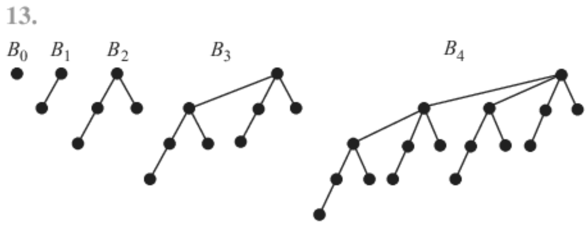
27. By Exercise 24, at each stage of Sollin's algorithm a forest results. Hence, after $n - 1$ edges are chosen, a tree results. It remains to show that this tree is a minimum spanning tree. Let $T$ be a minimum spanning tree with as many edges in common with Sollin's tree $S$ as possible. If $T \neq S$, then there is an edge $e \in S - T$ added at some stage in the algorithm, where prior to that stage all edges in $S$ are also in $T$. $T \cup \{e\}$ contains a unique simple circuit. Find an edge $e' \in S - T$ and an edge $e'' \in T - S$ on this circuit and "adjacent" when viewing the trees of this stage as "supervertices." Then by the algorithm, $w(e') \leq w(e'')$. So replace $T$ by $T - \{e''\} \cup \{e'\}$ to produce a minimum spanning tree closer to $S$ than $T$ was. 29. Each of the $r$ trees is joined to at least one other tree by a new edge. Hence, there are at most $r/2$ trees in the result (each new tree contains two or more old trees). To accomplish this, we need to add $r - (r/2) = r/2$ edges. Because the number of edges added is integral, it is at least $\lceil r/2 \rceil$. 31. If $k \geq \log n$, then $n/2^k \leq 1$, so $\lceil n/2^k \rceil = 1$, so by Exercise 30 the algorithm is finished after at most $\log n$ iterations. 33. Suppose that a minimum spanning tree $T$ contains edge $e = uv$ that is the maximum weight edge in simple circuit $C$. Delete $e$ from $T$. This creates a forest with two components, one containing $u$ and the other containing $v$. Follow the edges of the path $C - e$, starting at $u$. At some point this path must jump from the component of $T - e$ containing $u$ to the component of $T - e$ containing $v$, say using edge $f$. This edge cannot be in $T$, because $e$ can be the only edge of $T$ joining the two components (otherwise there would be a simple circuit in $T$). Because $e$ is the edge of greatest weight in $C$, the

weight of $f$ is smaller. The tree formed by replacing $e$ by $f$ in $T$ therefore has smaller weight, a contradiction. 35. The reverse-delete algorithm must terminate and produce a spanning tree, because the algorithm never disconnects the graph and upon termination there can be no more simple circuits. The edge deleted at each stage of the algorithm must have been the edge of maximum weight in whatever circuits it was a part of. Therefore by Exercise 33 it cannot be in any minimum spanning tree. Since only edges that could not have been in any minimum spanning tree have been deleted, the result must be a minimum spanning tree.

## Supplementary Exercises

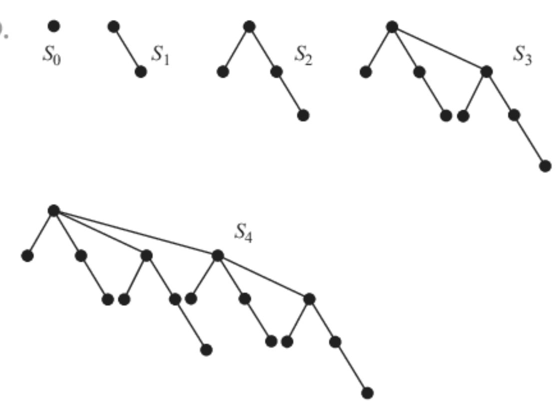
1. Suppose $T$ is a tree. Then clearly $T$ has no simple circuits. If we add an edge $e$ connecting two nonadjacent vertices $u$ and $v$, then obviously a simple circuit is formed, because when $e$ is added to $T$ the resulting graph has too many edges to be a tree. The only simple circuit formed is made up of the edge $e$ together with the unique path in $T$ from $v$ to $u$. Suppose $T$ satisfies the given conditions. All that is needed is to show that $T$ is connected, because there are no simple circuits in the graph. Assume that $T$ is not connected. Then let $u$ and $v$ be in separate connected components. Adding $e = \{u, v\}$ does not satisfy the conditions. 3. Suppose that a tree $T$ has $n$ vertices of degrees $d_1, d_2, \ldots, d_n$, respectively. Because $2e = \sum_{i=1}^{n} d_i$ and $e = n - 1$, we have $2(n-1) = \sum_{i=1}^{n} d_i$. Because each $d_i \geq 1$, it follows that $2(n-1) = n + \sum_{i=1}^{n}(d_i - 1)$, or that $n - 2 = \sum_{i=1}^{n}(d_i - 1)$. Hence, at most $n - 2$ of the terms of this sum can be 1 or more. Hence, at least two of them are 0. It follows that $d_i = 1$ for at least two values of $i$. 5. $2n - 2$ 7. A tree has no circuits, so it cannot have a subgraph homeomorphic to $K_{3,3}$ or $K_5$. 9. Color each connected component separately. For each of these connected components, first root the tree, then color all vertices at even levels red and all vertices at odd levels blue. 11. Upper bound: $k^h$; lower bound: $2\lceil k/2 \rceil^{h-1}$

13.




15. Because $B_{k+1}$ is formed from two copies of $B_k$, one shifted down one level, the height increases by 1 as $k$ increases by 1. Because $B_0$ had height 0, it follows by induction that $B_k$ has height $k$. 17. Because the root of $B_{k+1}$ is the root of $B_k$ with one additional child (namely the root of the other $B_k$), the degree of the root increases by 1 as $k$ increases by 1. Because $B_0$ had a root with degree 0, it follows by induction that $B_k$ has a root with degree $k$.

19.

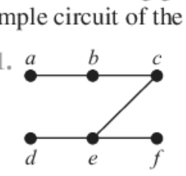



21. Use mathematical induction. The result is trivial for $k = 0$. Suppose it is true for $k - 1$. $T_{k-1}$ is the parent tree for $T$. By induction, the child tree for $T$ can be obtained from $T_0, \ldots, T_{k-2}$ in the manner stated. The final connection of $r_{k-2}$ to $r_{k-1}$ is as stated in the definition of $S_k$-tree.

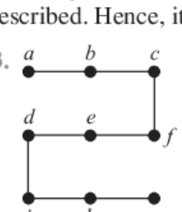
23. **procedure** *level*($T$: ordered rooted tree with root $r$)
    *queue* := sequence consisting of just the root $r$
    **while** *queue* contains at least one term
        $v$ := first vertex in queue
        list $v$
        remove $v$ from queue and put children of $v$ onto
          the end of queue

25. Build the tree by inserting a root for the address 0, and then inserting a subtree for each vertex labeled $i$, for $i$ a positive integer, built up from subtrees for each vertex labeled $i.j$ for $j$ a positive integer, and so on. 27. **a)** Yes **b)** No **c)** Yes 29. The resulting graph has no edge that is in more than one simple circuit of the type described. Hence, it is a cactus.

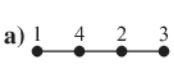
31.



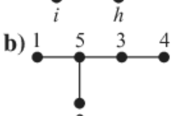

33.



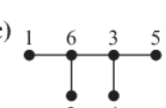

35. **a)**



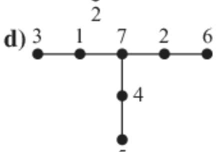

**b)**




**c)**




**d)**




37. 6 39. **a)** 0 for 00, 11 for 01, 100 for 10, 101 for 11 (exact coding depends on how ties were broken, but all versions are equivalent); $0.645n$ for string of length $n$ **b)** 0 for 000, 100 for 001, 101 for 010, 110 for 100, 11100 for 011, 11101 for 101, 11110 for 110, 11111 for 111 (exact coding depends on how ties were broken, but all versions are equivalent); $0.532\overline{6}n$ for string of length $n$ 41. Let $G'$ be the graph obtained by delet-

ing from $G$ the vertex $v$ and all edges incident to $v$. A minimum spanning tree of $G$ can be obtained by taking an edge of minimal weight incident to $v$ together with a minimum spanning tree of $G'$. 43. Suppose that edge $e$ is the edge of least weight incident to vertex $v$, and suppose that $T$ is a spanning tree that does not include $e$. Add $e$ to $T$, and delete from the simple circuit formed thereby the other edge of the circuit that contains $v$. The result will be a spanning tree of strictly smaller weight (because the deleted edge has weight greater than the weight of $e$). This is a contradiction, so $T$ must include $e$. 45. Because paths in trees are unique, an arborescence $T$ of a directed graph $G$ is just a subgraph of $G$ that is a tree rooted at $r$, containing all the vertices of $G$, with all the edges directed away from the root. Thus the in-degree of each vertex other than $r$ is 1. For the converse, it is enough to show that for each $v \in V$ there is a unique directed path from $r$ to $v$. Because the in-degree of each vertex other than $r$ is 1, we can follow the edges of $T$ backwards from $v$. This path can never return to a previously visited vertex, because that would create a simple circuit. Therefore the path must eventually stop, and it can stop only at $r$, whose in-degree is not necessarily 1. Following this path forward gives the path from $r$ to $v$ required by the definition of arborescence. 47. **a)** Run the breadth-first search algorithm, starting from $v$ and respecting the directions of the edges, marking each vertex encountered as reachable. **b)** Running breadth-first search on $G^{conv}$, again starting at $v$, respecting the directions of the edges, and marking each vertex encountered, will identify all the vertices from which $v$ is reachable. **c)** Choose a vertex $v_1$ and using parts (a) and (b) find the strong component containing $v_1$, namely all vertices $w$ such that $w$ is reachable from $v_1$ and $v_1$ is reachable from $w$. Then choose another vertex $v_2$ not yet in a strong component and find the strong component of $v_2$. Repeat until all vertices have been included. The correctness of this algorithm follows from the definition of strong component and Exercise 17 in Section 10.4.

# CHAPTER 12

## Section 12.1

1. **a)** 1 **b)** 1 **c)** 0 **d)** 0 3. **a)** $(1 \cdot 1) + (\overline{0 \cdot 1} + 0) = 1 + (\overline{0} + 0) = 1 + (1 + 0) = 1 + 1 = 1$ **b)** $(\mathbf{T} \wedge \mathbf{T}) \vee (\neg(\mathbf{F} \wedge \mathbf{T}) \vee \mathbf{F}) \equiv \mathbf{T}$

5. **a)**

| $x$ | $y$ | $z$ | $\overline{x}y$ |
|---|---|---|---|
| 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 |

**b)**

| $x$ | $y$ | $z$ | $x + yz$ |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 |

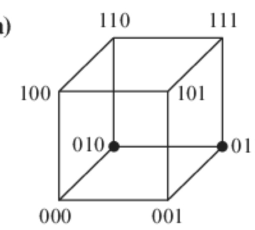
**c)**

| $x$ | $y$ | $z$ | $x\overline{y} + \overline{xyz}$ |
|---|---|---|---|
| 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

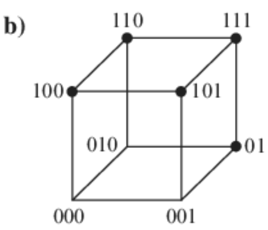
**d)**

| $x$ | $y$ | $z$ | $x(yz + \overline{y}\,\overline{z})$ |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 |

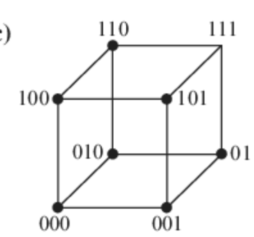
7. **a)**



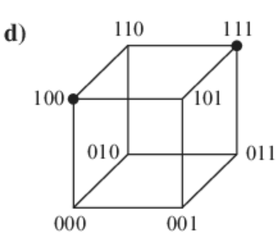

**b)**




**c)**




**d)**




9. (0, 0) and (1, 1) 11. $x + xy = x \cdot 1 + xy = x(1 + y) = x(y + 1) = x \cdot 1 = x$

13.

| $x$ | $y$ | $z$ | $x\overline{y}$ | $y\overline{z}$ | $\overline{x}z$ | $x\overline{y} + y\overline{z} + \overline{x}z$ | $\overline{x}y$ | $\overline{y}z$ | $x\overline{z}$ | $\overline{x}y + \overline{y}z + x\overline{z}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

15.

| $x$ | $x + x$ | $x \cdot x$ |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 1 | 1 |

17.

| $x$ | $x + 1$ | $x \cdot 0$ |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 0 |

19.

| $x$ | $y$ | $z$ | $y+z$ | $x+(y+z)$ | $x+y$ | $(x+y)+z$ | $yz$ | $x(yz)$ | $xy$ | $(xy)z$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

21.

| $x$ | $y$ | $xy$ | $\overline{(xy)}$ | $\overline{x}$ | $\overline{y}$ | $\overline{x}+\overline{y}$ | $x+y$ | $\overline{(x+y)}$ | $\overline{x}\,\overline{y}$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |

23. $0 \cdot \overline{0} = 0 \cdot 1 = 0$; $1 \cdot \overline{1} = 1 \cdot 0 = 0$

25.

| $x$ | $y$ | $x \oplus y$ | $x+y$ | $xy$ | $\overline{(xy)}$ | $(x+y)\overline{(xy)}$ | $x\overline{y}$ | $\overline{x}y$ | $x\overline{y}+\overline{x}y$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

27. **a)** True, as a table of values can show **b)** False; take $x = 1$, $y = 1$, $z = 1$, for instance **c)** False; take $x = 1$, $y = 1$, $z = 0$, for instance 29. By De Morgan's laws, the complement of an expression is like the dual except that the complement of each variable has been taken. 31. 16 33. If we replace each 0 by **F**, 1 by **T**, Boolean sum by $\vee$, Boolean product by $\wedge$, and $\overline{\ }$ by $\neg$ (and $x$ by $p$ and $y$ by $q$ so that the variables look like they represent propositions, and the equals sign by the logical equivalence symbol), then $\overline{xy} = \overline{x} + \overline{y}$ becomes $\neg(p \wedge q) \equiv \neg p \vee \neg q$ and $\overline{x+y} = \overline{x}\,\overline{y}$ becomes $\neg(p \vee q) \equiv \neg p \wedge \neg q$. 35. By the domination, distributive, and identity laws, $x \vee x = (x \vee x) \wedge 1 = (x \vee x) \wedge (x \vee \overline{x}) = x \vee (x \wedge \overline{x}) = x \vee 0 = x$. Similarly, $x \wedge x = (x \wedge x) \vee 0 = (x \wedge x) \vee (x \wedge \overline{x}) = x \wedge (x \vee \overline{x}) = x \wedge 1 = x$. 37. Because $0 \vee 1 = 1$ and $0 \wedge 1 = 0$ by the identity and commutative laws, it follows that $\overline{0} = 1$. Similarly, because $1 \vee 0 = 1$ and $1 \wedge 0 = 1$, it follows that $\overline{1} = 0$. 39. First, note that $x \wedge 0 = 0$ and $x \vee 1 = 1$ for all $x$, as can easily be proved. To prove the first identity, it is sufficient to show that $(x \vee y) \vee (\overline{x} \wedge \overline{y}) = 1$ and $(x \vee y) \wedge (\overline{x} \wedge \overline{y}) = 0$. By the associative, commutative, distributive, domination, and identity laws, $(x \vee y) \vee (\overline{x} \wedge \overline{y}) = y \vee [x \vee (\overline{x} \wedge \overline{y})] = y \vee [(x \vee \overline{x}) \wedge (x \vee \overline{y})] = y \vee [1 \wedge (x \vee \overline{y})] = y \vee (x \vee \overline{y}) = (y \vee \overline{y}) \vee x = 1 \vee x = 1$ and $(x \vee y) \wedge (\overline{x} \wedge \overline{y}) = \overline{y} \wedge [\overline{x} \wedge (x \vee y)] = \overline{y} \wedge [(\overline{x} \wedge x) \vee (\overline{x} \wedge y)] = \overline{y} \wedge [0 \vee (\overline{x} \wedge y)] = \overline{y} \wedge (\overline{x} \wedge y) = \overline{x} \wedge (y \wedge \overline{y}) = \overline{x} \wedge 0 = 0$. The second identity is proved in a similar way. 41. Using the hypotheses, Exercise 35, and the distributive law it follows that $x = x \vee 0 = x \vee (x \vee y) = (x \vee x) \vee y = x \vee y = 0$. Similarly, $y = 0$. To prove the second statement, note that $x = x \wedge 1 = x \wedge (x \wedge y) = (x \wedge x) \wedge y = x \wedge y = 1$. Similarly, $y = 1$. 43. Use Exercises 39 and 41 in the Supplementary Exercises in Chapter 9 and the definition of a complemented, distributed lattice to establish the five pairs of laws in the definition.

## Section 12.2

1. **a)** $\overline{x}\,\overline{y}z$ **b)** $\overline{x}y\overline{z}$ **c)** $\overline{x}yz$ **d)** $\overline{x}\,\overline{y}\,\overline{z}$ 3. **a)** $xyz + xy\overline{z} + x\overline{y}z + x\overline{y}\,\overline{z} + \overline{x}yz + \overline{x}y\overline{z} + \overline{x}\,\overline{y}z$ **b)** $xyz + xy\overline{z} + \overline{x}yz$ **c)** $xyz + xy\overline{z} + x\overline{y}z + x\overline{y}\,\overline{z}$ **d)** $x\overline{y}z + x\overline{y}\,\overline{z}$ 5. $wxy\overline{z} + wx\overline{y}z + w\overline{x}yz + \overline{w}xyz + \overline{w}x\overline{y}\,\overline{z} + \overline{w}\,\overline{x}\,\overline{y}z + \overline{w}\,\overline{x}y\overline{z} + w\overline{x}\,\overline{y}\,\overline{z}$ 7. **a)** $\overline{x} + \overline{y} + z$ **b)** $x + y + z$ **c)** $x + \overline{y} + z$ 9. $y_1 + y_2 + \cdots + y_n = 0$ if and only if $y_i = 0$ for $i = 1, 2, \ldots, n$. This holds if and only if $x_i = 0$ when $y_i = x_i$ and $x_i = 1$ when $y_i = \overline{x}_i$. 11. **a)** $x + y + z$ **b)** $(x + y + z)(x + y + \overline{z})(x + \overline{y} + z)(\overline{x} + y + z)(\overline{x} + y + \overline{z})$ **c)** $(x + y + z)(x + y + \overline{z})(x + \overline{y} + z)(x + \overline{y} + \overline{z})$ **d)** $(x + y + z)(x + y + \overline{z})(x + \overline{y} + z)(x + \overline{y} + \overline{z})(\overline{x} + \overline{y} + z)(\overline{x} + \overline{y} + \overline{z})$ 13. **a)** $x + y + z$ **b)** $x + [y + \overline{(\overline{x} + z)}]$ **c)** $\overline{(x + \overline{y})}$ **d)** $[x + \overline{(x + \overline{y} + \overline{z})}]$

15. **a)**

| $x$ | $\overline{x}$ | $x \downarrow x$ |
|---|---|---|
| 1 | 0 | 0 |
| 0 | 1 | 1 |

**b)**

| $x$ | $y$ | $xy$ | $x \downarrow x$ | $y \downarrow y$ | $(x \downarrow x) \downarrow (y \downarrow y)$ |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 0 |

**c)**

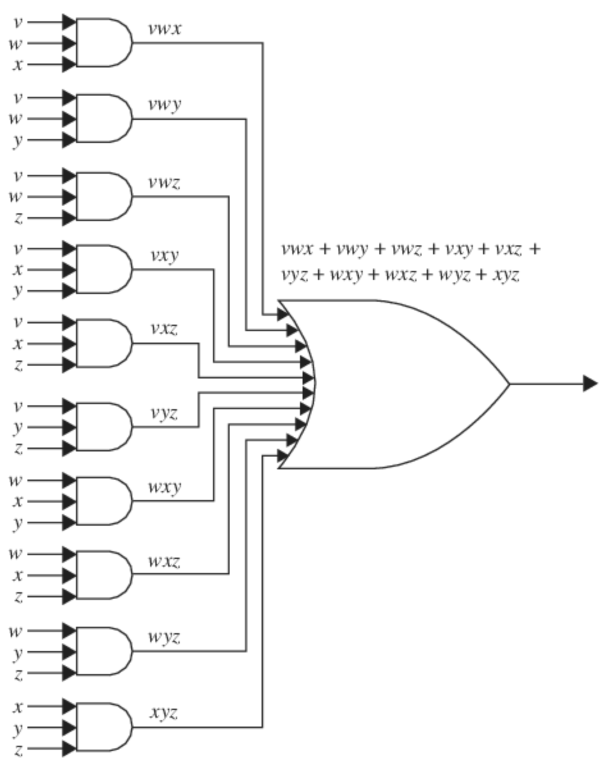
| $x$ | $y$ | $x+y$ | $(x \downarrow y)$ | $(x \downarrow y) \downarrow (x \downarrow y)$ |
|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |

17. **a)** $\{[(x \mid x) \mid (y \mid y)] \mid [(x \mid x) \mid (y \mid y)]\} \mid (z \mid z)$ **b)** $\{[(x \mid x) \mid (z \mid z)] \mid y\} \mid \{[(x \mid x) \mid (z \mid z)] \mid y\}$ **c)** $x$ **d)** $[x \mid (y \mid y)] \mid [x \mid (y \mid y)]$ 19. It is impossible to represent $\overline{x}$ using $+$ and $\cdot$ because there is no way to get the value 0 if the input is 1.

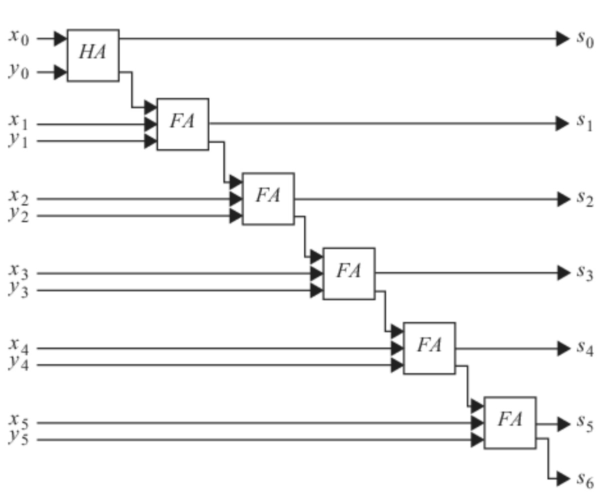
## Section 12.3

1. $(x + y)\overline{y}$ 3. $\overline{(xy)} + (\overline{z} + x)$ 5. $(x + y + z) + (\overline{x} + y + z) + (\overline{x} + \overline{y} + \overline{z})$

**7.**

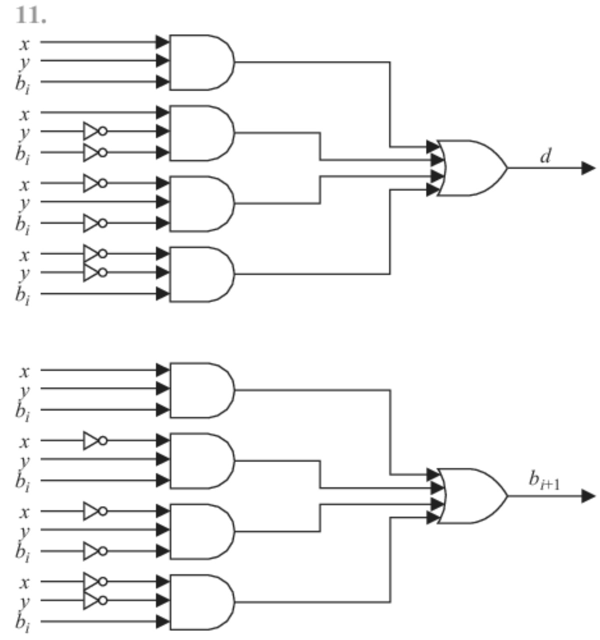



**9.**

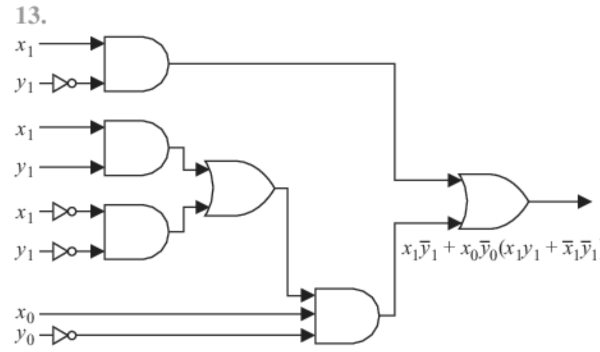



*HA* = half adder
*FA* = full adder

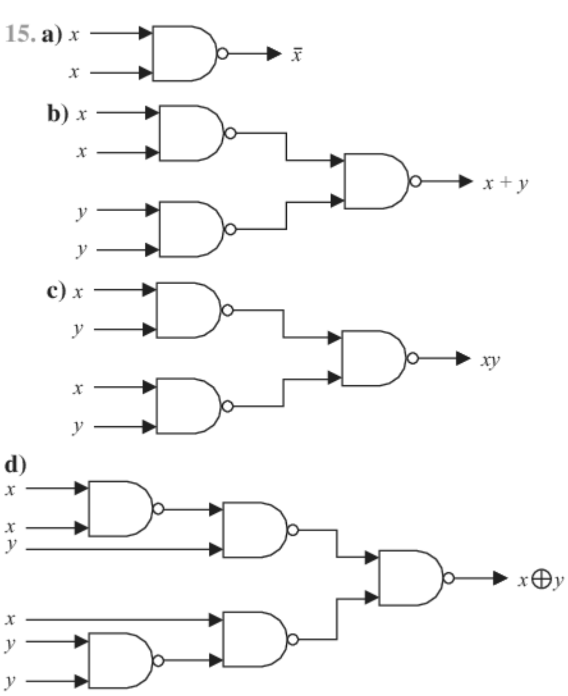
**11.**



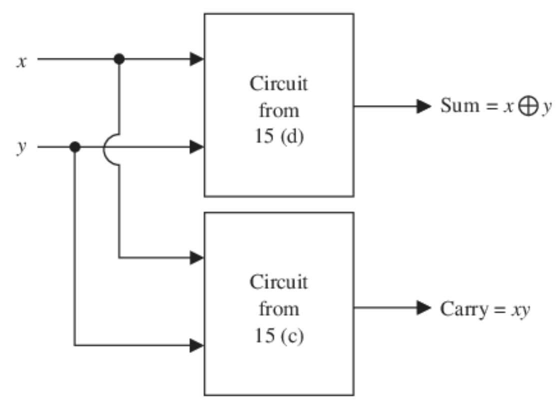

**13.**



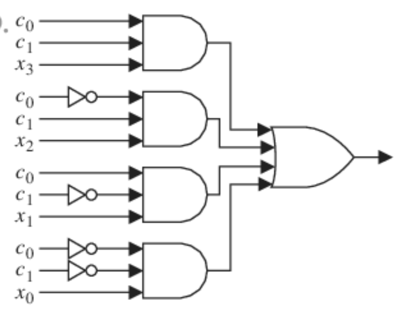

**15.**



**17.**




**19.**




## Section 12.4

**1. a)**

| | $y$ | $\overline{y}$ |
|---|---|---|
| $x$ | | |
| $\overline{x}$ | 1 | |

**b)** $xy$ and $\overline{x}\,\overline{y}$

**3. a)**

| | $y$ | $\overline{y}$ |
|---|---|---|
| $x$ | | 1 |
| $\overline{x}$ | | |

**b)**

| | $y$ | $\overline{y}$ |
|---|---|---|
| $x$ | 1 | |
| $\overline{x}$ | | 1 |

**c)**

| | $y$ | $\overline{y}$ |
|---|---|---|
| $x$ | 1 | 1 |
| $\overline{x}$ | 1 | 1 |

**5. a)**

| | $yz$ | $y\overline{z}$ | $\overline{y}\,\overline{z}$ | $\overline{y}z$ |
|---|---|---|---|---|
| $x$ | | | | |
| $\overline{x}$ | | 1 | | |

**b)** $\overline{x}yz$, $\overline{x}\,\overline{y}\,\overline{z}$, $xy\overline{z}$

**7. a)**

| | $yz$ | $y\overline{z}$ | $\overline{y}\,\overline{z}$ | $\overline{y}z$ |
|---|---|---|---|---|
| $x$ | | | 1 | |
| $\overline{x}$ | | | | |

**b)**

| | $yz$ | $y\overline{z}$ | $\overline{y}\,\overline{z}$ | $\overline{y}z$ |
|---|---|---|---|---|
| $x$ | | | | |
| $\overline{x}$ | 1 | | 1 | |

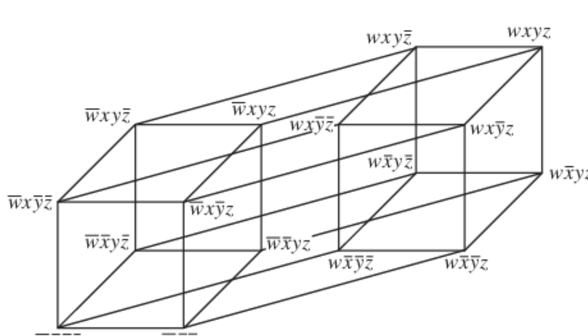
**c)**

| | $yz$ | $y\overline{z}$ | $\overline{y}\,\overline{z}$ | $\overline{y}z$ |
|---|---|---|---|---|
| $x$ | 1 | 1 | | 1 |
| $\overline{x}$ | | | | 1 |

**9.** Implicants: $xyz$, $xy\overline{z}$, $x\,\overline{y}\,\overline{z}$, $\overline{x}y\overline{z}$, $xy$, $x\overline{z}$, $y\overline{z}$; prime implicants: $xy$, $x\overline{z}$, $y\overline{z}$; essential prime implicants: $xy$, $x\overline{z}$, $y\overline{z}$

| | $yz$ | $y\overline{z}$ | $\overline{y}\,\overline{z}$ | $\overline{y}z$ |
|---|---|---|---|---|
| $x$ | 1 | 1 | 1 | |
| $\overline{x}$ | | 1 | | |

**11.** The 3-cube on the right corresponds to $w$; the 3-cube given by the top surface of the whole figure represents $x$; the 3-cube given by the back surface of the whole figure represents $y$; the 3-cube given by the right surfaces of both the left and the right 3-cube represents $z$. In each case, the opposite 3-face represents the complemented literal. The 2-cube that represents $wz$ is the right face of the 3-cube on the right; the 2-cube that represents $\overline{x}y$ is bottom rear; the 2-cube that represents $\overline{y}\,\overline{z}$ is front left.



**13. a)**

| | $yz$ | $y\bar{z}$ | $\bar{y}\bar{z}$ | $\bar{y}z$ |
|---|---|---|---|---|
| $wx$ | | | | |
| $w\bar{x}$ | | | | |
| $\bar{w}\bar{x}$ | | | | |
| $\bar{w}x$ | | 1 | | |

**b)** $\bar{w}xyz$, $\bar{w}\bar{x}y\bar{z}$, $\bar{w}x\bar{y}\bar{z}$, $wxy\bar{z}$

**15. a)**

| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | 1 | 1 | | | | | | |
| $x_1\bar{x}_2$ | | | | | | | | |
| $\bar{x}_1\bar{x}_2$ | | | | | | | | |
| $\bar{x}_1x_2$ | | | | | | | | |

**b)**

| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | | | | | | | | |
| $x_1\bar{x}_2$ | | | | | | | | |
| $\bar{x}_1\bar{x}_2$ | 1 | | | 1 | | | | |
| $\bar{x}_1x_2$ | 1 | | | 1 | | | | |

**c)**

| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | 1 | 1 | | | | | 1 | 1 |
| $x_1\bar{x}_2$ | | | | | | | | |
| $\bar{x}_1\bar{x}_2$ | | | | | | | | |
| $\bar{x}_1x_2$ | 1 | 1 | | | | | 1 | 1 |

**d)**

| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | | | | | 1 | 1 | | |
| $x_1\bar{x}_2$ | | | | | 1 | 1 | | |
| $\bar{x}_1\bar{x}_2$ | | | | | 1 | 1 | | |
| $\bar{x}_1x_2$ | | | | | 1 | 1 | | |

**e)**

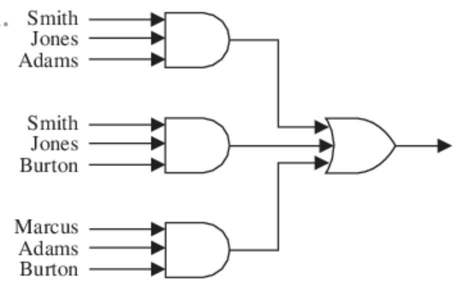
| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | 1 | 1 | 1 | 1 | | | | |
| $x_1\bar{x}_2$ | 1 | 1 | 1 | 1 | | | | |
| $\bar{x}_1\bar{x}_2$ | 1 | 1 | 1 | 1 | | | | |
| $\bar{x}_1x_2$ | 1 | 1 | 1 | 1 | | | | |

**f)**

| | $x_3x_4x_5$ | $x_3x_4\bar{x}_5$ | $x_3\bar{x}_4\bar{x}_5$ | $x_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4x_5$ | $\bar{x}_3\bar{x}_4\bar{x}_5$ | $\bar{x}_3x_4\bar{x}_5$ | $\bar{x}_3x_4x_5$ |
|---|---|---|---|---|---|---|---|---|
| $x_1x_2$ | | 1 | 1 | | | 1 | 1 | |
| $x_1\bar{x}_2$ | | 1 | 1 | | | 1 | 1 | |
| $\bar{x}_1\bar{x}_2$ | | 1 | 1 | | | 1 | 1 | |
| $\bar{x}_1x_2$ | | 1 | 1 | | | 1 | 1 | |

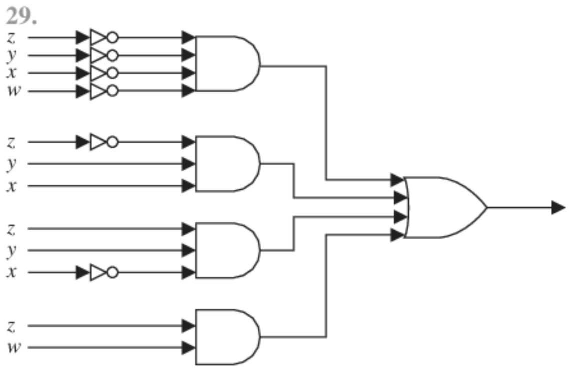
**17. a)** 64 **b)** 6 **19.** Rows 1 and 4 are considered adjacent. The pairs of columns considered adjacent are: columns 1 and 4, 1 and 12, 1 and 16, 2 and 11, 2 and 15, 3 and 6, 3 and 10, 4 and 9, 5 and 8, 5 and 16, 6 and 15, 7 and 10, 7 and 14, 8 and 13, 9 and 12, 11 and 14, 13 and 16.

**21.**




**23. a)** $\bar{x}z$ **b)** $y$ **c)** $x\bar{z} + \bar{x}z + \bar{y}z$ **d)** $xz + \bar{x}y + \bar{y}\bar{z}$ **25. a)** $wxz + wx\bar{y} + w\bar{y}z + w\bar{x}\bar{y}\bar{z}$ **b)** $x\bar{y}z + \bar{w}\bar{y}z + wxy\bar{z} + w\bar{x}yz + \bar{w}\bar{x}y\bar{z}$ **c)** $\bar{y}z + wxy + w\bar{x}\bar{y} + \bar{w}\bar{x}y\bar{z}$ **d)** $wy + yz + \bar{x}y + wxz + \bar{w}\bar{x}\bar{z}$ **27.** $x(y+z)$

29.




31. $\overline{x}\,\overline{z}+xz$ 33. We use induction on $n$. If $n=1$, then we are looking at a line segment, labeled 0 at one end and 1 at the other end. The only possible value of $k$ is also 1, and if the literal is $x_1$, then the subcube we have is the 0-dimensional subcube consisting of the endpoint labeled 1, and if the literal is $\overline{x}_1$, then the subcube we have is the 0-dimensional subcube consisting of the endpoint labeled 0. Now assume that the statement is true for $n$; we must show that it is true for $n+1$. If the literal $x_{n+1}$ (or its complement) is not part of the product, then by the inductive hypothesis, the product when viewed in the setting of $n$ variables corresponds to an $(n-k)$-dimensional subcube of the $n$-dimensional cube, and the Cartesian product of that subcube with the line segment [0, 1] gives us a subcube one dimension higher in our given $(n+1)$-dimensional cube, namely having dimension $(n+1)-k$, as desired. On the other hand, if the literal $x_{n+1}$ (or its complement) is part of the product, then the product of the remaining $k-1$ literals corresponds to a subcube of dimension $n-(k-1)=(n+1)-k$ in the $n$-dimensional cube, and that slice, at either the 1-end or the 0-end in the last variable, is the desired subcube.

## Supplementary Exercises

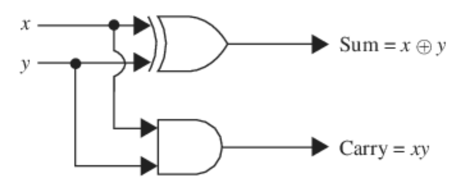
1. **a)** $x=0,\ y=0,\ z=0$; $x=1,\ y=1,\ z=1$ **b)** $x=0,\ y=0,\ z=0$; $x=0,\ y=0,\ z=1$; $x=0,\ y=1,\ z=0$; $x=1,\ y=0,\ z=1$; $x=1,\ y=1,\ z=0$; $x=1,\ y=1,\ z=1$ **c)** No values 3. **a)** Yes **b)** No **c)** No **d)** Yes 5. $2^{2^{n-1}}$ 7. **a)** If $F(x_1,\ldots,x_n)=1$, then $(F+G)(x_1,\ldots,x_n)=F(x_1,\ldots,x_n)+G(x_1,\ldots,x_n)=1$ by the dominance law. Hence, $F\le F+G$. **b)** If $(FG)(x_1,\ldots,x_n)=1$, then $F(x_1,\ldots,x_n)\cdot G(x_1,\ldots,x_n)=1$. Hence, $F(x_1,\ldots,x_n)=1$. It follows that $FG\le F$. 9. Because $F(x_1,\ldots,x_n)=1$ implies that $F(x_1,\ldots,x_n)=1$, $\le$ is reflexive. Suppose that $F\le G$ and $G\le F$. Then $F(x_1,\ldots,x_n)=1$ if and only if $G(x_1,\ldots,x_n)=1$. This implies that $F=G$. Hence, $\le$ is antisymmetric. Suppose that $F\le G\le H$. Then if $F(x_1,\ldots,x_n)=1$, it follows that $G(x_1,\ldots,x_n)=1$, which implies that $H(x_1,\ldots,x_n)=1$. Hence, $F\le H$, so $\le$ is transitive. 11. **a)** $x=1,\ y=0,\ z=0$ **b)** $x=1,\ y=0,\ z=0$ **c)** $x=1,\ y=0,\ z=0$

13.

| $x$ | $y$ | $x \odot y$ | $x \oplus y$ | $\overline{(x \oplus y)}$ |
|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |

15. Yes, as a truth table shows 17. **a)** 6 **b)** 5 **c)** 5 **d)** 6

19.




21. $x_3+x_2\overline{x}_1$ 23. Suppose it were with weights $a$ and $b$. Then there would be a real number $T$ such that $xa+yb\ge T$ for (1,0) and (0,1), but with $xa+yb<T$ for (0,0) and (1,1). Hence, $a\ge T$, $b\ge T$, $0<T$, and $a+b<T$. Thus, $a$ and $b$ are positive, which implies that $a+b>a\ge T$, a contradiction.

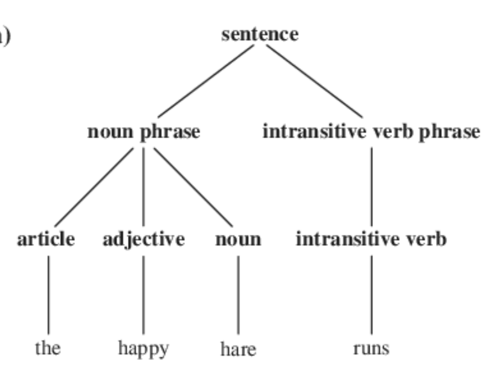
# CHAPTER 13

## Section 13.1

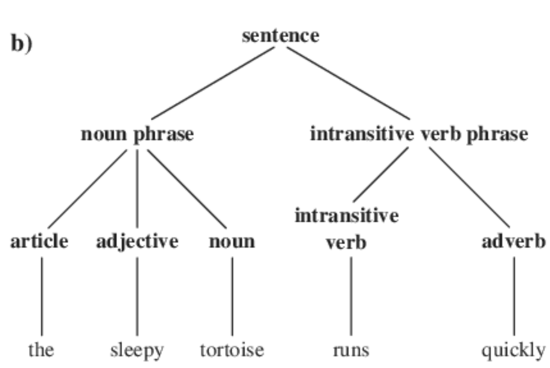
1. **a) sentence** ⇒ **noun phrase intransitive verb phrase** ⇒ **article adjective noun intransitive verb phrase** ⇒ **article adjective noun intransitive verb** ⇒ . . . (after 3 steps) . . . ⇒ *the happy hare runs*.
**b) sentence** ⇒ **noun phrase intransitive verb phrase** ⇒ **article adjective noun intransitive verb phrase** ⇒ **article adjective noun intransitive verb adverb**…(after 4 steps)…⇒ *the sleepy tortoise runs quickly*
**c) sentence** ⇒ **noun phrase transitive verb phrase noun phrase** ⇒ **article noun transitive verb phrase noun phrase** ⇒ **article noun transitive verb noun phrase** ⇒ **article noun transitive verb article noun** ⇒ . . . (after 4 steps)…⇒ *the tortoise passes the hare*
**d) sentence** ⇒ **noun phrase transitive verb phrase noun phrase** ⇒ **article adjective noun transitive verb phrase noun phrase** ⇒ **article adjective noun transitive verb noun phrase** ⇒ **article adjective noun transitive verb article adjective noun** ⇒ . . . (after 6 steps)…⇒ *the sleepy hare passes the happy tortoise*
3. The only way to get a noun, such as *tortoise*, at the end is to have a noun phrase at the end, which can be achieved only via the production **sentence** → **noun phrase transitive verb phrase noun phrase**. However, **transitive verb phrase** → **transitive verb** → *passes*, and this sentence does not contain *passes*.
5. **a)** $S \Rightarrow 1A \Rightarrow 10B \Rightarrow 101A \Rightarrow 1010B \Rightarrow 10101$ **b)** Because of the productions in this grammar, every 1 must be followed by a 0 unless it occurs at the end of the string. **c)** All strings consisting of a 0 or a 1 followed by one or more repetitions of 01

7. $S \Rightarrow 0S1 \Rightarrow 00S11 \Rightarrow 000S111 \Rightarrow 000111$ 9. **a)** $S \Rightarrow 0S \Rightarrow 00S \Rightarrow 00S1 \Rightarrow 00S11 \Rightarrow 00S111 \Rightarrow 00S1111 \Rightarrow 001111$ **b)** $S \Rightarrow 0S \Rightarrow 00S \Rightarrow 001A \Rightarrow 0011A \Rightarrow 00111A \Rightarrow 001111$ 11. $S \Rightarrow 0SAB \Rightarrow 00SABAB \Rightarrow 00ABAB \Rightarrow 00\ AABB \Rightarrow 001ABB \Rightarrow 0011BB \Rightarrow 00112B \Rightarrow 001122$ 13. **a)** $S \rightarrow 0$, $S \rightarrow 1$, $S \rightarrow 11$ **b)** $S \rightarrow 1S$, $S \rightarrow \lambda$ **c)** $S \rightarrow 0A1$, $A \rightarrow 1A$, $A \rightarrow 0A$, $A \rightarrow \lambda$ **d)** $S \rightarrow 0A$, $A \rightarrow 11A$, $A \rightarrow \lambda$ 15. **a)** $S \rightarrow 00S$, $S \rightarrow \lambda$ **b)** $S \rightarrow 10A$, $A \rightarrow 00A$, $A \rightarrow \lambda$ **c)** $S \rightarrow AAS$, $S \rightarrow BBS$, $AB \rightarrow BA$, $BA \rightarrow AB$, $S \rightarrow \lambda$, $A \rightarrow 0$, $B \rightarrow 1$ **d)** $S \rightarrow 0000000000A$, $A \rightarrow 0A$, $A \rightarrow \lambda$ **e)** $S \rightarrow AS$, $S \rightarrow ABS$, $S \rightarrow A$, $AB \rightarrow BA$, $BA \rightarrow AB$, $A \rightarrow 0$, $B \rightarrow 1$ **f)** $S \rightarrow ABS$, $S \rightarrow \lambda$, $AB \rightarrow BA$, $BA \rightarrow AB$, $A \rightarrow 0$, $B \rightarrow 1$ **g)** $S \rightarrow ABS$, $S \rightarrow T$, $S \rightarrow U$, $T \rightarrow AT$, $T \rightarrow A$, $U \rightarrow BU$, $U \rightarrow B$, $AB \rightarrow BA$, $BA \rightarrow AB$, $A \rightarrow 0$, $B \rightarrow 1$ 17. **a)** $S \rightarrow 0S$, $S \rightarrow \lambda$ **b)** $S \rightarrow A0$, $A \rightarrow 1A$, $A \rightarrow \lambda$ **c)** $S \rightarrow 000S$, $S \rightarrow \lambda$ 19. **a)** Type 2, not type 3 **b)** Type 3 **c)** Type 0, not type 1 **d)** Type 2, not type 3 **e)** Type 2, not type 3 **f)** Type 0, not type 1 **g)** Type 3 **h)** Type 0, not type 1 **i)** Type 2, not type 3 **j)** Type 2, not type 3 21. Let $S_1$ and $S_2$ be the start symbols of $G_1$ and $G_2$, respectively. Let $S$ be a new start symbol. **a)** Add $S$ and productions $S \rightarrow S_1$ and $S \rightarrow S_2$. **b)** Add $S$ and production $S \rightarrow S_1 S_2$. **c)** Add $S$ and production $S \rightarrow \lambda$ and $S \rightarrow S_1 S$.

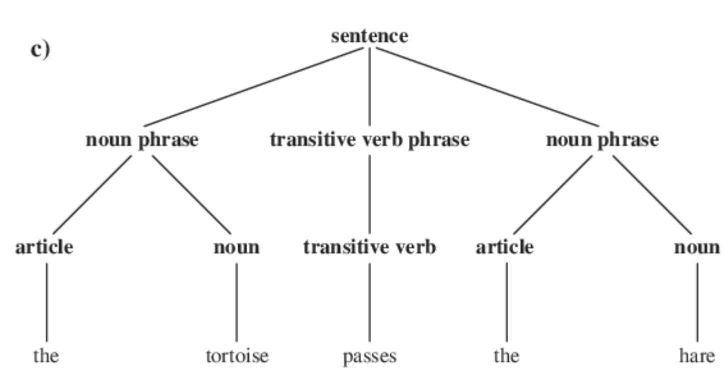
23. **a)**



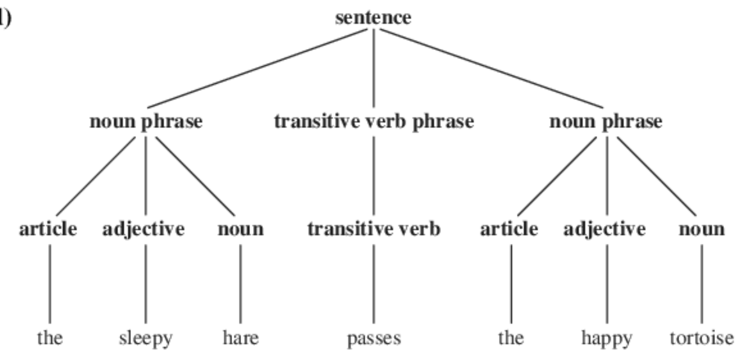

**b)**




**c)**

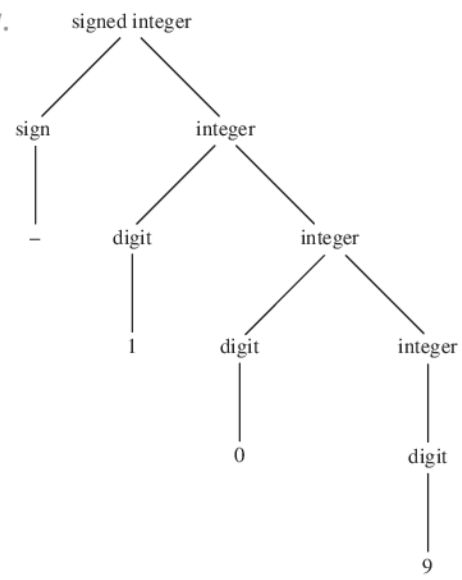


**d)**




**25. a)** Yes **b)** No **c)** Yes **d)** No

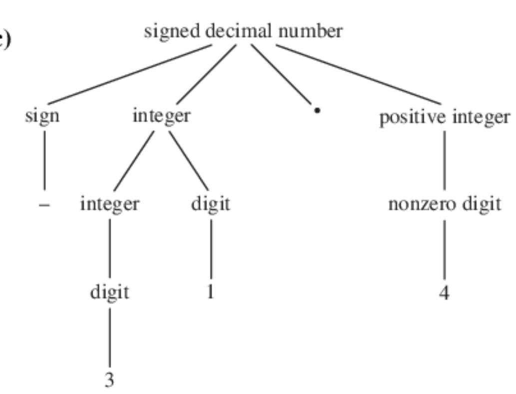
**27.**




**29. a)** $S \to \langle sign\rangle\langle integer\rangle$
$S \to \langle sign\rangle\langle integer\rangle\ .\ \langle positive\ integer\rangle$
$\langle sign\rangle \to +$
$\langle sign\rangle \to -$
$\langle integer\rangle \to \langle digit\rangle$
$\langle integer\rangle \to \langle integer\rangle\langle digit\rangle$
$\langle digit\rangle \to i,\ i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 0$
$\langle positive\ integer\rangle \to \langle integer\rangle\langle nonzero\ digit\rangle$
$\langle positive\ integer\rangle \to \langle nonzero\ digit\rangle\langle integer\rangle$
$\langle positive\ integer\rangle \to \langle integer\rangle\langle nonzero\ digit\rangle\langle integer\rangle$
$\langle positive\ integer\rangle \to \langle nonzero\ digit\rangle$
$\langle nonzero\ digit\rangle \to i,\ i = 1, 2, 3, 4, 5, 6, 7, 8, 9$

**b)** $\langle signed\ decimal\ number\rangle ::= \langle sign\rangle\langle integer\rangle \mid \langle sign\rangle\langle integer\rangle\ .\ \langle positive\ integer\rangle$
$\langle sign\rangle ::= + \mid -$
$\langle integer\rangle ::= \langle digit\rangle \mid \langle integer\rangle\langle digit\rangle$
$\langle digit\rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$
$\langle nonzero\ digit\rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$
$\langle positive\ integer\rangle ::= \langle integer\rangle\langle nonzero\ digit\rangle \mid \langle nonzero\ digit\rangle\langle integer\rangle \mid \langle integer\rangle\langle nonzero\ integer\rangle\langle integer\rangle \mid \langle nonzero\ digit\rangle$

**c)**



**31. a)** $\langle identifier \rangle ::= \langle lcletter \rangle \mid \langle identifier \rangle\langle lcletter \rangle$
$\langle lcletter \rangle ::= a \mid b \mid c \mid \cdots \mid z$
**b)** $\langle identifier \rangle ::= \langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle \mid \langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle \mid$
$\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle \mid$
$\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle\langle lcletter \rangle$
$\langle lcletter \rangle ::= a \mid b \mid c \mid \cdots \mid z$
**c)** $\langle identifier \rangle ::= \langle ucletter \rangle \mid \langle ucletter \rangle\langle letter \rangle \mid \langle ucletter \rangle\langle letter \rangle\langle letter \rangle \mid$
$\langle ucletter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle \mid \langle ucletter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle \mid$
$\langle ucletter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle\langle letter \rangle$
$\langle letter \rangle ::= \langle lcletter \rangle \mid \langle ucletter \rangle$
$\langle lcletter \rangle ::= a \mid b \mid c \mid \cdots \mid z$
$\langle ucletter \rangle ::= A \mid B \mid C \mid \cdots \mid Z$
**d)** $\langle identifier \rangle ::= \langle lcletter \rangle\langle digitorus \rangle\langle alphanumeric \rangle\langle alphanumeric \rangle\langle alphanumeric \rangle \mid$
$\langle lcletter \rangle\langle digitorus \rangle\langle alphanumeric \rangle\langle alphanumeric \rangle\langle alphanumeric \rangle\langle alphanumeric \rangle$
$\langle digitorus \rangle ::= \langle digit \rangle \mid \_$
$\langle alphanumeric \rangle ::= \langle letter \rangle \mid \langle digit \rangle$
$\langle letter \rangle ::= \langle lcletter \rangle \mid \langle ucletter \rangle$
$\langle lcletter \rangle ::= a \mid b \mid c \mid \cdots \mid z$
$\langle ucletter \rangle ::= A \mid B \mid C \mid \cdots \mid Z$
$\langle digit \rangle ::= 0 \mid 1 \mid 2 \mid \cdots \mid 9$

**33.** $\langle identifier \rangle ::= \langle letterorus \rangle \mid \langle identifier \rangle\langle symbol \rangle$
$\langle letterorus \rangle ::= \langle letter \rangle \mid \_$
$\langle symbol \rangle ::= \langle letterorus \rangle \mid \langle digit \rangle$
$\langle letter \rangle ::= \langle lcletter \rangle \mid \langle ucletter \rangle$
$\langle lcletter \rangle ::= a \mid b \mid c \mid \cdots \mid z$
$\langle ucletter \rangle ::= A \mid B \mid C \mid \cdots \mid Z$
$\langle digit \rangle ::= 0 \mid 1 \mid 2 \mid \cdots \mid 9$

**35.** $numeral ::= sign?\ nonzerodigit\ digit{*}\ decimal? \mid sign?\ 0\ decimal?$
$sign ::= + \mid -$
$nonzerodigit ::= 1 \mid 2 \mid \cdots \mid 9$
$digit ::= 0 \mid nonzerodigit$
$decimal ::= .digit{*}$

**37.** $identifier ::= letterorus\ symbol{*}$
$letterorus ::= letter \mid \_$
$symbol ::= letterorus \mid digit$
$letter ::= lcletter \mid ucletter$
$lcletter ::= a \mid b \mid c \mid \cdots \mid z$
$ucletter ::= A \mid B \mid C \mid \cdots \mid Z$
$digit ::= 0 \mid 1 \mid 2 \mid \cdots \mid 9$

**39. a)** $\langle expression \rangle$
$\langle term \rangle\langle term \rangle\langle addOperator \rangle$
$\langle factor \rangle\langle factor \rangle\langle factor \rangle\langle mulOperator \rangle\langle addOperator \rangle$
$\langle identifier \rangle\langle identifier \rangle\langle identifier \rangle\langle mulOperator \rangle\langle addOperator \rangle$
$a\ b\ c * +$
**b)** Not generated
**c)** $\langle expression \rangle$
$\langle term \rangle$
$\langle factor \rangle\langle factor \rangle\langle mulOperator \rangle$
$\langle expression \rangle\langle factor \rangle\langle mulOperator \rangle$
$\langle term \rangle\langle term \rangle\langle addOperator \rangle\langle factor \rangle\langle mulOperator \rangle$
$\langle factor \rangle\langle factor \rangle\langle addOperator \rangle\langle factor \rangle\langle mulOperator \rangle$
$\langle identifier \rangle\langle identifier \rangle\langle addOperator \rangle\langle identifier \rangle\langle mulOperator \rangle$
$x\ y - z\ *$

**d)** ⟨*expression*⟩

⟨*term*⟩

⟨*factor*⟩⟨*factor*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*expression*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*term*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*factor*⟩⟨*factor*⟩⟨*mulOperator*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*factor*⟩⟨*expression*⟩⟨*mulOperator*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*factor*⟩⟨*term*⟩⟨*term*⟩⟨*addOperator*⟩⟨*mulOperator*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*factor*⟩⟨*factor*⟩⟨*factor*⟩⟨*addOperator*⟩⟨*mulOperator*⟩⟨*mulOperator*⟩

⟨*identifier*⟩⟨*identifier*⟩⟨*identifier*⟩⟨*identifier*⟩⟨*addOperator*⟩⟨*mulOperator*⟩⟨*mulOperator*⟩

$w\ x\ y\ z - * /$

**e)** ⟨*expression*⟩

⟨*term*⟩

⟨*factor*⟩⟨*factor*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*expression*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*term*⟩⟨*term*⟩⟨*addOperator*⟩⟨*mulOperator*⟩

⟨*factor*⟩⟨*factor*⟩⟨*factor*⟩⟨*addOperator*⟩⟨*mulOperator*⟩

⟨*identifier*⟩⟨*identifier*⟩⟨*identifier*⟩⟨*addOperator*⟩⟨*mulOperator*⟩

$a\ d\ e - *$

41. **a)** Not generated

**b)** ⟨*expression*⟩

⟨*term*⟩⟨*addOperator*⟩⟨*term*⟩

⟨*factor*⟩⟨*mulOperator*⟩⟨*factor*⟩⟨*addOperator*⟩⟨*factor*⟩⟨*mulOperator*⟩⟨*factor*⟩

⟨*identifier*⟩⟨*mulOperator*⟩⟨*identifier*⟩⟨*addOperator*⟩⟨*identifier*⟩⟨*mulOperator*⟩⟨*identifier*⟩

$a/b + c/d$

**c)** ⟨*expression*⟩

⟨*term*⟩

⟨*factor*⟩⟨*mulOperator*⟩⟨*factor*⟩

⟨*factor*⟩⟨*mulOperator*⟩(⟨*expression*⟩)

⟨*factor*⟩⟨*mulOperator*⟩(⟨*term*⟩⟨*addOperator*⟩⟨*term*⟩)

⟨*factor*⟩⟨*mulOperator*⟩(⟨*factor*⟩⟨*addOperator*⟩⟨*factor*⟩)

⟨*identifier*⟩⟨*mulOperator*⟩(⟨*identifier*⟩⟨*addOperator*⟩⟨*identifier*⟩)

$m * (n + p)$

**d)** Not generated

**e)** ⟨*expression*⟩

⟨*term*⟩

⟨*factor*⟩⟨*mulOperator*⟩⟨*factor*⟩

(⟨*expression*⟩)⟨*mulOperator*⟩(⟨*expression*⟩)

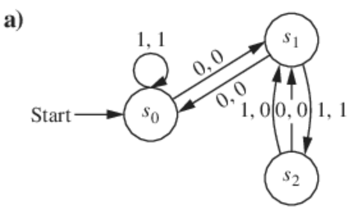
(⟨*term*⟩⟨*addOperator*⟩⟨*term*⟩)⟨*mulOperator*⟩(⟨*term*⟩⟨*addOperator*⟩⟨*term*⟩)    (⟨*factor*⟩⟨*addOperator*⟩⟨*factor*⟩)⟨*mulOperator*⟩(⟨*factor*⟩⟨*addOp*

(⟨*identifier*⟩⟨*addOperator*⟩⟨*identifier*⟩)⟨*mulOperator*⟩(⟨*identifier*⟩⟨*addOperator*⟩⟨*identifier*⟩)

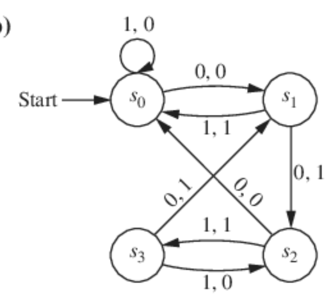
$(m + n) * (p - q)$

## Section 13.2

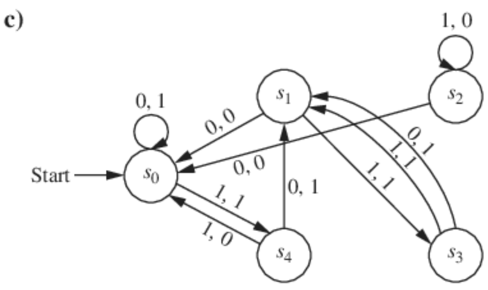
**1. a)**




**b)**

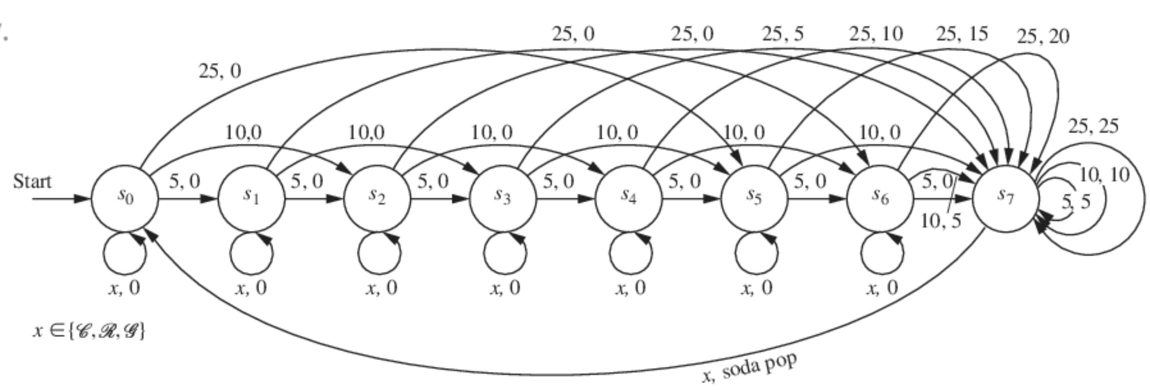



**c)**



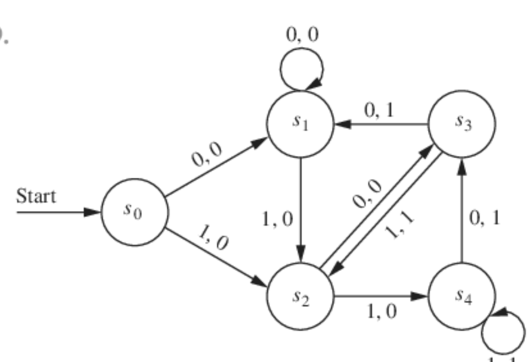

**3. a)** 01010 **b)** 01000 **c)** 11011 **5. a)** 1100 **b)** 00110110 **c)** 11111111111

**7.**

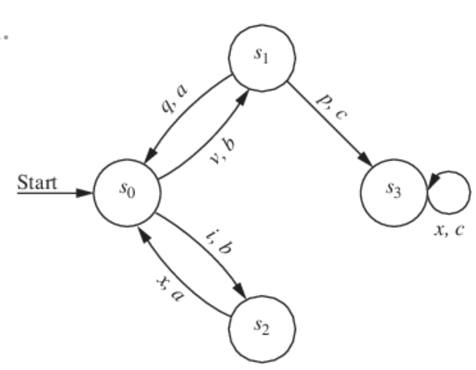



$\mathscr{C}$ = cola
$\mathscr{R}$ = root beer
$\mathscr{G}$ = ginger ale

**9.**

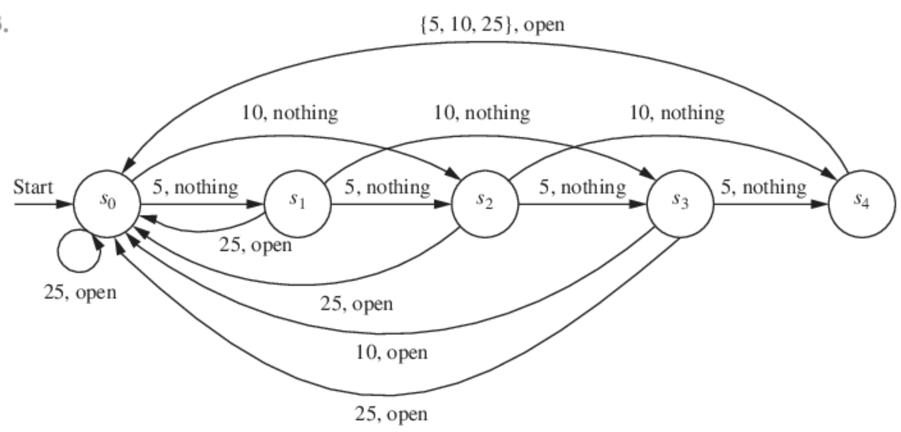



**11.**



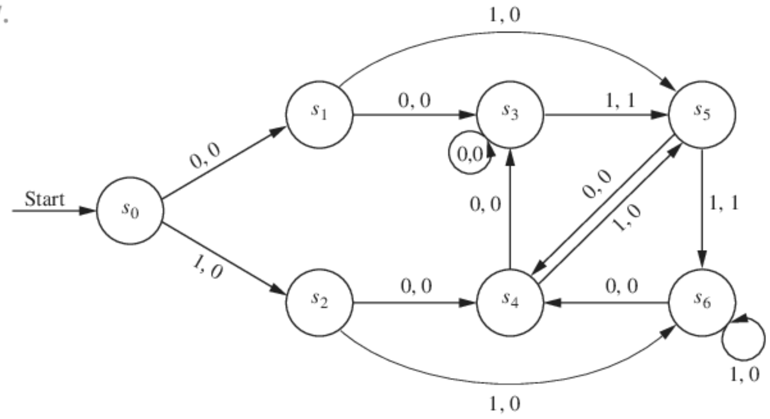

$v$ = Valid ID $a$ = "Enter user ID"
$i$ = Invalid ID $b$ = "Enter password"
$p$ = Valid password $c$ = Prompt
$q$ = Invalid password $x$ = Any input

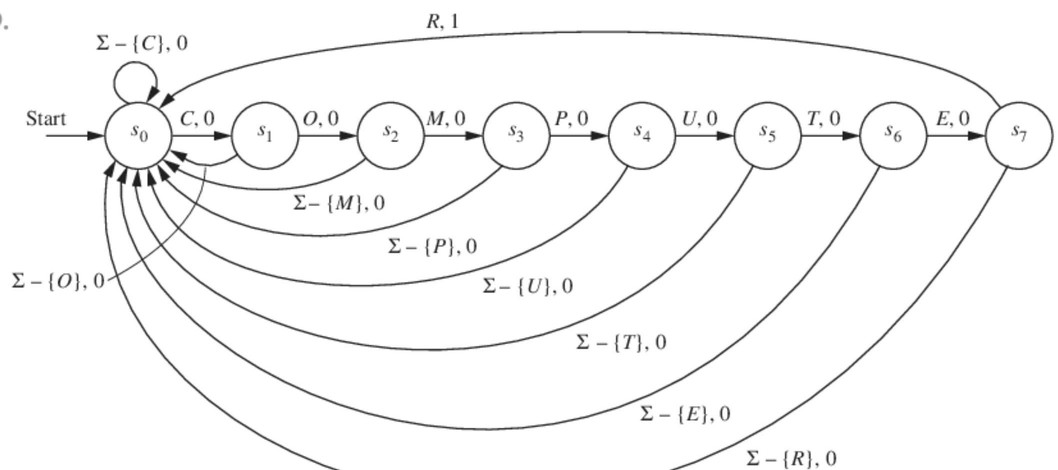
**13.**




**15.** Let $s_0$ be the start state and let $s_1$ be the state representing a successful call. From $s_0$, inputs of 2, 3, 4, 5, 6, 7, or 8 send the machine back to $s_0$ with output of an error message for the user. From $s_0$ an input of 0 sends the machine to state $s_1$, with the output being that the 0 is sent to the network. From $s_0$ an input of 9 sends the machine to state $s_2$ with no output; from there an input of 1 sends the machine to state $s_3$ with no output; from there an input of 1 sends the machine to state $s_1$ with the output being that the 911 is sent to the network. All other inputs while in states $s_2$ or $s_3$ send the machine back to $s_0$ with output of an error message for the user. From $s_0$ an input of 1 sends the machine to state $s_4$ with no output; from $s_4$ an input of 2 sends the machine to state $s_5$ with no output; and this path continues in a similar manner to the 911 path, looking next for 1, then 2, then any seven digits, at which point the machine goes to state $s_1$ with the output being that the ten-digit input is sent to the network. Any "incorrect" input while in states $s_5$ or $s_6$ (that is, anything except a 1 while in $s_5$ or a 2 while in $s_6$) sends the machine back to $s_0$ with output of an error message for the user. Similarly, from $s_4$ an input of 8 followed by appropriate successors drives us eventually to $s_1$, but inappropriate outputs drive us back to $s_0$ with an error message. Also, inputs while in state $s_4$ other than 2 or 8 send the machine back to state $s_0$ with output of an error message for the user.

**17.**

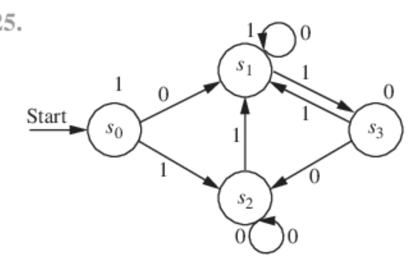


19.



21.

| | *f* | | |
|---|---|---|---|
| | ***Input*** | | |
| ***State*** | **0** | **1** | ***g*** |
| $s_0$ | $s_1$ | $s_2$ | 1 |
| $s_1$ | $s_1$ | $s_0$ | 1 |
| $s_2$ | $s_1$ | $s_2$ | 0 |

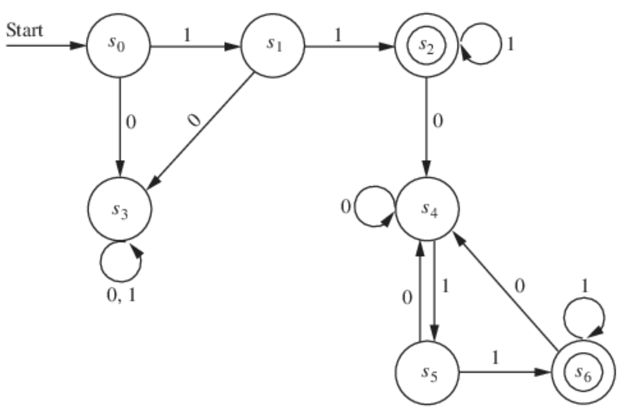
23. **a)** 11111 **b)** 1000000 **c)** 100011001100

25.



## Section 13.3

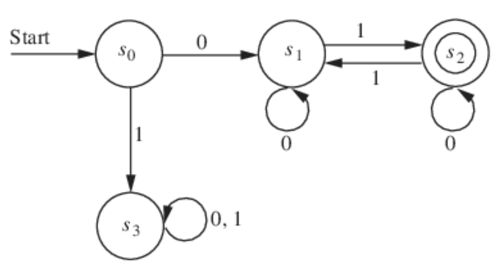
1. **a)** $\{000, 001, 1100, 1101\}$ **b)** $\{000, 0011, 010, 0111\}$ **c)** $\{00, 011, 110, 1111\}$ **d)** $\{000000, 000001, 000100, 000101, 010000, 010001, 010100, 010101\}$ 3. $A = \{1, 101\}$, $B = \{0, 11, 000\}$; $A = \{10, 111, 1010, 1000, 10111, 101000\}$, $B = \{\lambda\}$; $A = \{\lambda, 10\}$, $B = \{10, 111, 1000\}$ or $A = \{\lambda\}$, $B = \{10, 111, 1010, 1000, 10111, 101000\}$ 5. **a)** The set of strings consisting of zero or more consecutive bit pairs 10 **b)** The set of strings consisting of all 1s such that the number of 1s is divisible by 3, including the null string **c)** The set of strings in which every 1 is immediately preceded by a 0 **d)** The set of strings that begin and end with a 1 and have at least two 1s between every pair of 0s 7. A string is in $A^*$ if and only if it is a concatenation of an arbitrary number of strings in $A$. Because each string in $A$ is also in $B$, it follows that a string in $A^*$ is also a concatenation of strings in $B$. Hence, $A^* \subseteq B^*$. 9. **a)** Yes **b)** Yes **c)** No **d)** No **e)** Yes **f)** Yes 11. **a)** Yes **b)** No **c)** Yes **d)** No 13. **a)** Yes **b)** Yes **c)** No **d)** No **e)** No **f)** No 15. We use structural induction on the input string $y$. The basis step considers $y = \lambda$, and for the inductive step we write $y = wa$, where $w \in I^*$ and $a \in I$. For the basis step, we have $xy = x$, so we must show that $f(s, x) = f(f(s, x), \lambda)$. But part $(i)$ of the definition of the extended transition function says that this is true. We then assume the inductive hypothesis that the equation holds for $w$ and prove that $f(s, xwa) = f(f(s, x), wa)$. By part $(ii)$ of the definition, the left-hand side of this equation equals $f(f(s, xw), a)$. By the inductive hypothesis, $f(s, xw) = f(f(s, x), w)$, so $f(f(s, xw), a) = f(f(f(s, x), w), a)$. The right-hand side of our desired equality is, by part $(ii)$ of the definition, also equal to $f(f(f(s, x), w), a)$, as desired. 17. $\{0, 10, 11\}\{0, 1\}^*$ 19. $\{0^m 1^n \mid m \geq 0 \text{ and } n \geq 1\}$ 21. $\{\lambda\} \cup \{0\}\{1\}^*\{0\} \cup \{10, 11\}\{0, 1\}^* \cup \{0\}\{1\}^*\{01\}\{0, 1\}^* \cup \{0\}\{1\}^*\{00\}\{0\}^*\{1\}\{0, 1\}^*$ 23. Let $s_2$ be the only final state, and put transitions from $s_2$ to itself on either input. Put a transition from the start state $s_0$ to $s_1$ on input 0, and a transition from $s_1$ to $s_2$ on input 1. Create state $s_3$, and have the other transitions from $s_0$ and $s_1$ (as well as both transitions from $s_3$) lead to $s_3$. 25. Start state $s_0$, only final state $s_3$; transitions from $s_0$ to $s_0$ on 0, from $s_0$ to $s_1$ on 1, from $s_1$ to $s_2$ on 0, from $s_1$ to $s_1$ on 1, from $s_2$ to $s_0$ on 0, from $s_2$ to $s_3$ on 1, from $s_3$ to $s_3$ on 0, from $s_3$ to $s_3$ on 1 27. Have five states, with only $s_3$ final. For $i = 0, 1, 2, 3$, transition from $s_i$ to itself on input 1 and to $s_{i+1}$ on input 0. Both transitions from $s_4$ are to itself. 29. Have four states, with only $s_3$ final. For $i = 0, 1, 2$, transition from $s_i$ to $s_{i+1}$ on input 1 but back to $s_0$ on input 0. Both transitions from $s_3$ are to itself.

**31.**




**33.** Start state $s_0$, only final state $s_1$; transitions from $s_0$ to $s_0$ on 1, from $s_0$ to $s_1$ on 0, from $s_1$ to $s_1$ on 1; from $s_1$ to $s_0$ on 0

**35.**



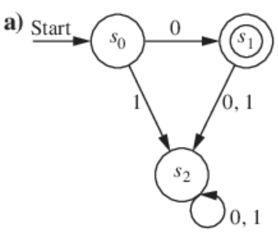

**37.** Suppose that such a machine exists, with start state $s_0$ and other state $s_1$. Because the empty string is not in the language but some strings are accepted, we must have $s_1$ as the only final state, with at least one transition from $s_0$ to $s_1$. Because the string 0 is not in the language, the transition from $s_0$ on input 0 must be to itself, so the transition from $s_0$ on input 1 must be to $s_1$. But this contradicts the fact that 1 is not in the language. **39.** Change each final state to a nonfinal state and vice versa. **41.** Same machine as in Exercise 25, but with $s_0$, $s_1$, and $s_2$ as the final states **43.** $\{0, 01, 11\}$ **45.** $\{\lambda, 0\} \cup \{0^m 1^n \mid m \geq 1, n \geq 1\}$ **47.** $\{10^n \mid n \geq 0\} \cup \{10^n 10^m \mid n, m \geq 0\}$ **49.** The union of the set of all strings that start with a 0 and the set of all strings that have no 0s

**51.**

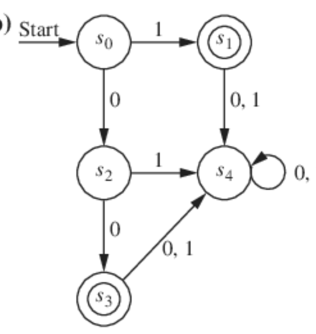



**53.** Add a nonfinal state $s_3$ with transitions to $s_3$ from $s_0$ on input 0, from $s_1$ on input 1, and from $s_3$ on input 0 or 1.

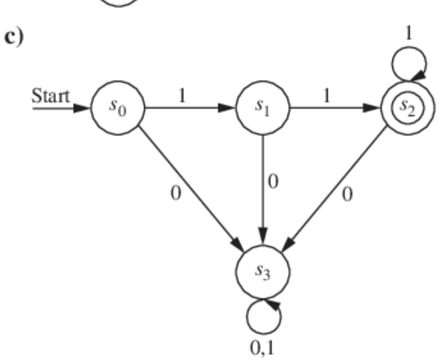
**55. a)**




**b)**




**c)**



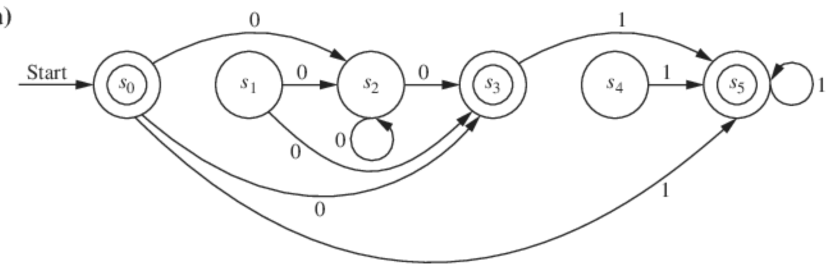

**57.** Suppose that $M$ is a finite-state automaton that accepts the set of bit strings containing an equal number of 0s and 1s. Suppose $M$ has $n$ states. Consider the string $0^{n+1}1^{n+1}$. By the pigeonhole principle, as $M$ processes this string, it must encounter the same state more than once as it reads the first $n+1$ 0s; so let $s$ be a state it hits at least twice. Then $k$ 0s in the input takes $M$ from state $s$ back to itself for some positive integer $k$. But then $M$ ends up exactly at the same place after reading $0^{n+1+k}1^{n+1}$ as it will after reading $0^{n+1}1^{n+1}$. Therefore, because $M$ accepts $0^{n+1}1^{n+1}$ it also accepts $0^{n+k+1}1^{n+1}$, which is a contradiction. **59.** We know from Exercise 58d that the equivalence classes of $R_k$ are a refinement of the equivalence classes of $R_{k-1}$ for each positive integer $k$. The equivalence classes are finite sets, and finite sets cannot be refined indefinitely (the most refined they can be is for each equivalence class to contain just one state). Therefore this sequence of refinements must remain unchanged from some point onward. It remains to show that as soon as we have $R_n = R_{n+1}$, then $R_n = R_m$ for all $m > n$, from which it follows that $R_n = R_*$, and so the equivalence classes for these two relations will be the same. By induction, it suffices to show that if $R_n = R_{n+1}$, then $R_{n+1} = R_{n+2}$. Suppose that $R_{n+1} \neq R_{n+2}$. This means that there are states $s$ and $t$ that are $(n+1)$-equivalent but not

$(n+2)$-equivalent. Thus there is a string $x$ of length $n+2$ such that, say, $f(s,x)$ is final but $f(t,x)$ is nonfinal. Write $x = aw$, where $a \in I$. Then $f(s,a)$ and $f(t,a)$ are not $(n+1)$-equivalent, because $w$ drives the first to a final state and the second to a nonfinal state. But $f(s,a)$ and $f(t,a)$ are $n$-equivalent, because $s$ and $t$ are $(n+1)$-equivalent. This contradicts the fact that $R_n = R_{n+1}$. **61. a)** By the way the machine $\overline{M}$ was constructed, a string will drive $M$ from the start state to a final state if and only if that string drives $\overline{M}$ from the start state to a final state. **b)** For a proof of this theorem, see a source such as *Introduction to Automata Theory, Languages, and Computation* (2nd Edition) by John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman (Addison-Wesley, 2000).

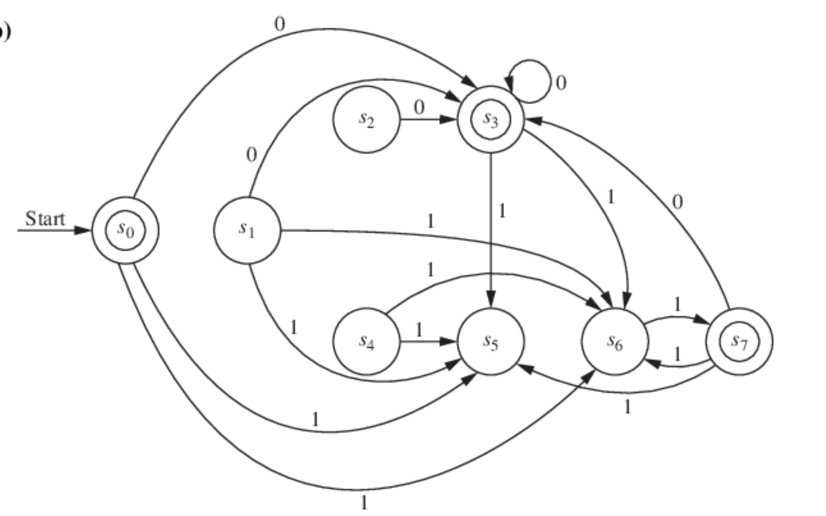
## Section 13.4

**1. a)** Any number of 1s followed by a 0 **b)** Any number of 1s followed by one or more 0s **c)** 111 or 001 **d)** A string of any number of 1s or 00s or some of each in a row **e)** $\lambda$ or a string that ends with a 1 and has one or more 0s before each 1 **f)** A string of length at least 3 that ends with 00 **3. a)** No **b)** No **c)** Yes **d)** Yes **e)** Yes **f)** No **g)** No **h)** Yes **5. a)** $\mathbf{0 \cup 11 \cup 010}$ **b)** $\mathbf{000000^*}$ **c)** $\mathbf{(0 \cup 1)((0 \cup 1)(0 \cup 1))^*}$ **d)** $\mathbf{0^*10^*}$ **e)** $\mathbf{(1 \cup 01 \cup 001)^*}$ **7. a)** $\mathbf{00^*1}$ **b)** $\mathbf{(0 \cup 1)(0 \cup 1)(0 \cup 1)^*0000^*}$ **c)** $\mathbf{0^*1^* \cup 1^*0^*}$ **d)** $\mathbf{11(111)^*(00)^*}$ **9. a)** Have the start state $s_0$, nonfinal, with no transitions. **b)** Have the start state $s_0$, final, with no transitions. **c)** Have the nonfinal start state $s_0$ and a final state $s_1$ and the transition from $s_0$ to $s_1$ on input $a$. **11.** Use an inductive proof. If the regular expression for $A$ is $\emptyset$, $\lambda$, or $\boldsymbol{x}$, the result is trivial. Otherwise, suppose the regular expression for $A$ is **BC**. Then $A = BC$ where $B$ is the set generated by **B** and $C$ is the set generated by **C**. By the inductive hypothesis there are regular expressions $\mathbf{B}'$ and $\mathbf{C}'$ that generate $B^R$ and $C^R$, respectively. Because $A^R = (BC)^R = C^R B^R$, $\mathbf{C}'\mathbf{B}'$ is a regular expression for $A^R$. If the regular expression for $A$ is $\mathbf{B} \cup \mathbf{C}$, then the regular expression for $A^R$ is $\mathbf{B}' \cup \mathbf{C}'$ because $(B \cup C)^R = (B^R) \cup (C^R)$. Finally, if the regular expression for $A$ is $\mathbf{B}^*$, then it is easy to see that $(\mathbf{B}')^*$ is a regular expression for $A^R$.

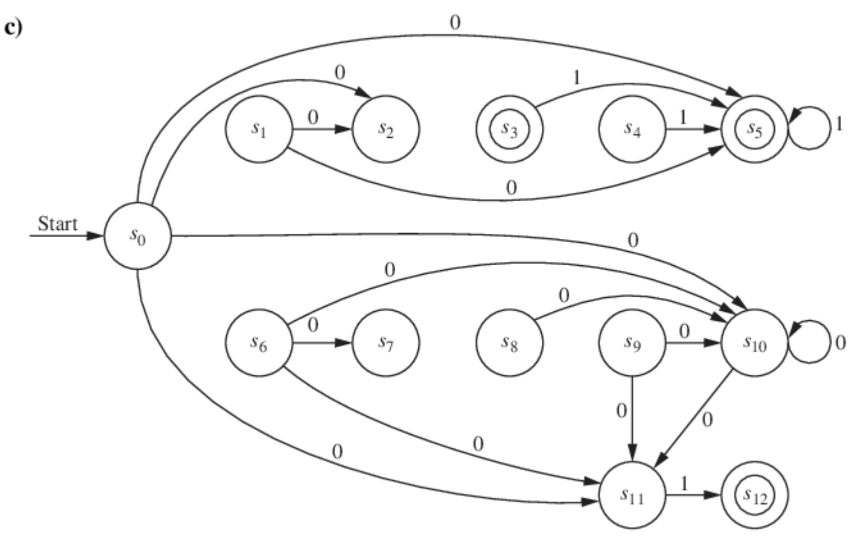
**13. a)**



**b)**




**c)**




**15.** $S \to 0A$, $S \to 1B$, $S \to 0$, $A \to 0B$, $A \to 1B$, $B \to 0B$, $B \to 1B$ **17.** $S \to 0C$, $S \to 1A$, $S \to 1$, $A \to 1A$, $A \to 0C$, $A \to 1$, $B \to 0B$, $B \to 1B$, $B \to 0$, $B \to 1$, $C \to 0C$, $C \to 1B$, $C \to 1$. **19.** This follows because input that leads to a final state in the automaton corresponds uniquely to a derivation in the grammar. **21.** The "only if" part is clear because $I$ is finite. For the "if" part let the states be $s_{i_0}, s_{i_1}, s_{i_2}, \ldots, s_{i_n}$, where $n = l(x)$. Because $n \geq |S|$, some state is repeated by the pigeonhole principle. Let $y$ be the part of $x$ that causes the loop, so that $x = uyv$ and $y$ sends $s_j$ to $s_j$, for some $j$. Then $uy^kv \in L(M)$ for all $k$. Hence, $L(M)$ is infinite. **23.** Suppose that $L = \{0^{2n}1^n, n = 0, 1, 2 \ldots\}$ were regular. Let $S$ be the set of states of a finite-state machine recognizing this set. Let $z = 0^{2n}1^n$ where $3n \geq |S|$. Then by the pumping lemma, $z = 0^{2n}1^n = uvw$, $l(v) \geq 1$, and $uv^iw \in \{0^{2n}1^n \mid n \geq 0\}$. Obviously $v$ cannot contain both 0 and 1, because $v^2$ would then contain 10. So $v$ is all 0s or all 1s, and hence, $uv^2w$ contains too many 0s or too many 1s, so it is not in $L$. This contradiction shows that $L$ is not regular. **25.** Suppose that the set of palindromes over $\{0, 1\}$ were regular. Let $S$ be the set of states of a finite-state machine recognizing this set. Let $z = 0^n10^n$, where $n > |S|$. Apply the pumping lemma to get $uv^iw \in L$ for all nonnegative integers $i$ where $l(v) \geq 1$, and $l(uv) \leq |S|$, and $z = 0^n10^n = uvw$. Then $v$ must be a string of 0s (because $n > |S|$), so $uv^2w$ is not a palindrome. Hence, the set of palindromes is not regular. **27.** Let $z = 1$; then $111 \notin L$ but $101 \in L$, so 11 and 10 are distinguishable. For the second question, the only way for $1z$ to be in $L$ is for $z$ to end with 01, and that is also the only way for $11z$ to be in $L$, so 1 and 11 are indistinguishable. **29.** This follows immediately from Exercise 28, because the $n$ distinguishable strings must drive the machine from the start state to $n$ different states. **31.** Any two distinct strings of the same length are distinguishable with respect to the language $P$ of all palindromes, because if $x$ and $y$ are distinct strings of length $n$, then $xx^R \in P$ but $yx^R \notin P$. Because there are $2^n$ different strings of length $n$, Exercise 29 tells us that any deterministic finite-state automaton for recognizing palindromes must have at least $2^n$ states. Because $n$ is arbitrary, this is impossible.

## Section 13.5

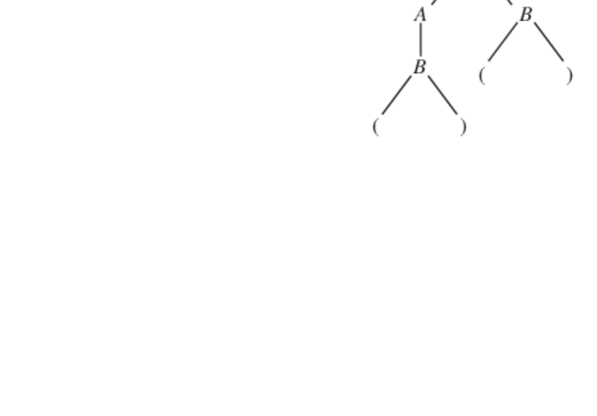
1. **a)** The nonblank portion of the tape contains the string 1111 when the machine halts. **b)** The nonblank portion of the tape contains the string 011 when the machine halts. **c)** The nonblank portion of the tape contains the string 00001 when the machine halts. **d)** The nonblank portion of the tape contains the string 00 when the machine halts. 3. **a)** The machine halts (and accepts) at the blank following the input, having changed the tape from 11 to 01. **b)** The machine changes every other occurrence of a 1, if any, starting with the first, to a 0, and otherwise leaves the string unchanged; it halts (and accepts) when it comes to the end of the string. 5. **a)** Halts with 01 on the tape, and does not accept **b)** The first 1 (if any) is changed to a 0 and the others are left alone. The input is not accepted.
7. $(s_0, 0, s_1, 1, R)$, $(s_0, 1, s_0, 1, R)$ 9. $(s_0, 0, s_0, 0, R)$, $(s_0, 1, s_1, 1, R)$, $(s_1, s_1, 0, R)$, $(s_1, 1, s_1, 0, R)$ 11. $(s_0, 0, s_1, 0, R)$, $(s_0, 1, s_0, 0, R)$, $(s_1, 0, s_1, 0, R)$, $(s_1, 1, s_0, 0, R)$, $(s_1, B, s_2, B, R)$ 13. $(s_0, 0, s_0, 0, R)$, $(s_0, 1, s_1, 1, R)$, $(s_1, 0, s_1, 0, R)$, $(s_1, 1, s_0, 1, R)$, $(s_0, B, s_2, B, R)$ 15. If the input string is blank or starts with a 1 the machine halts in nonfinal state $s_0$. Otherwise, the initial 0 is changed to an $M$ and the machine skips past all the intervening 0s and 1s until it either comes to the end of the input string or else comes to an $M$. At this point, it backs up one square and enters state $s_2$. Because the acceptable strings must have a 1 at the right for each 0 at the left, there must be a 1 here if the string is acceptable. Therefore, the only transition out of $s_2$ occurs when this square contains a 1. If it does, the machine replaces it with an $M$ and makes its way back to the left; if it does not, the machine halts in nonfinal state $s_2$. On its way back, it stays in $s_3$ as long as it sees 1s, then stays in $s_4$ as long as it sees 0s. Eventually either it encouters a 1 while in $s_4$ at which point it halts without accepting or else it reaches the rightmost $M$ that had been written over a 0 at the start of the string. If it is in $s_3$ when this happens, then there are no more 0s in the string, so it had better be the case that there are no more 1s either; this is accomplished by the transitions $(s_3, M, s_5, M, R)$ and $(s_5, M, s_6, M, R)$, and $s_6$ is a final state. Otherwise the machine halts in nonfinal state $s_5$. If it is in $s_4$ when this $M$ is encountered, things start all over again, except now the string will have had its leftmost remaining 0 and its rightmost remaining 1 replaced by $M$s. So the machine moves, staying in state $s_4$, to the leftmost remaining 0 and goes back into state $s_0$ to repeat the process.

17. $(s_0, B, s_9, B, L)$, $(s_0, 0, s_1, 0, L)$, $(s_1, B, s_2, E, R)$, $(s_2, M, s_2, M, R)$, $(s_2, 0, s_3, M, R)$, $(s_3, 0, s_3, 0, R)$, $(s_3, M, s_3, M, R)$, $(s_3, 1, s_4, M, R)$, $(s_4, 1, s_4, 1, R)$, $(s_4, M, s_4, M, R)$, $(s_4, 2, s_5, M, R)$, $(s_5, 2, s_5, 2, R)$, $(s_5, B, s_6, B, L)$, $(s_6, M, s_8, M, L)$, $(s_6, 2, s_7, 2, L)$, $(s_7, 0, s_7, 0, L)$, $(s_7, 1, s_7, 1, L)$, $(s_7, 2, s_7, 2, L)$, $(s_7, M, s_7, M, L)$, $(s_7, E, s_2, E, R)$, $(s_8, M, s_8, M, L)$, $(s_8, E, s_9, E, L)$ where $M$ and $E$ are markers, with $E$ marking the left end of the input
19. $(s_0, 1, s_1, B, R)$, $(s_1, 1, s_2, B, R)$, $(s_2, 1, s_3, B, R)$, $(s_3, 1, s_4, 1, R)$, $(s_1, B, s_4, 1, R)$, $(s_2, B, s_4, 1, R)$, $(s_3, B, s_4, 1, R)$
21. $(s_0, 1, s_1, B, R)$, $(s_1, 1, s_2, B, R)$, $(s_1, B, s_6, B, R)$, $(s_2, 1, s_3, B, R)$, $(s_2, B, s_6, B, R)$, $(s_3, 1, s_4, B, R)$, $(s_3, B, s_6, B, R)$, $(s_4, 1, s_5, B, R)$, $(s_4, B, s_6, B, R)$, $(s_6, B, s_{10}, 1, R)$, $(s_5, 1, s_5, B, R)$, $(s_5, B, s_7, 1, R)$, $(s_7, B, s_8, 1, R)$, $(s_8, B, s_9, 1, R)$, $(s_9, B, s_{10}, 1, R)$
23. $(s_0, 1, s_0, 1, R)$, $(s_0, B, s_1, B, L)$, $(s_1, 1, s_2, 0, L)$, $(s_2, 0, s_2, 0, L)$, $(s_2, 1, s_3, 0, R)$, $(s_2, B, s_6, B, R)$, $(s_3, 0, s_3, 0, R)$, $(s_3, 1, s_3, 1, R)$, $(s_3, B, s_4, 1, R)$, $(s_4, B, s_5, 1, L)$, $(s_5, 1, s_5, 1, L)$, $(s_5, 0, s_2, 0, L)$, $(s_6, 0, s_6, 1, R)$, $(s_6, 1, s_7, 1, R)$, $(s_6, B, s_7, B, R)$
25. $(s_0, 0, s_0, 0, R)$, $(s_0, *, s_5, B, R)$, $(s_3, *, s_3, *, L)$, $(s_3, 0, s_3, 0, L)$, $(s_3, 1, s_3, 1, L)$, $(s_3, B, s_0, B, R)$, $(s_5, 1, s_5, B, R)$, $(s_5, 0, s_5, B, R)$, $(s_5, B, s_6, B, L)$, $(s_6, B, s_6, B, L)$, $(s_6, 0, s_7, 1, L)$, $(s_7, 0, s_7, 1, L)$, $(s_0, 1, s_1, 0, R)$, $(s_1, 1, s_1, 1, R)$, $(s_1, *, s_2, *, R)$, $(s_2, 0, s_2, 0, R)$, $(s_2, 1, s_3, 0, L)$, $(s_2, B, s_4, B, L)$, $(s_4, 0, s_4, 1, L)$, $(s_4, *, s_8, B, L)$, $(s_8, 0, s_8, B, L)$, $(s_8, 1, s_8, B, L)$
27. Suppose that $s_m$ is the only halt state for the Turing machine in Exercise 22, where $m$ is the largest state number, and suppose that we have designed that machine so that when the machine halts the tape head is reading the leftmost 1 of the answer. Renumber each state in the machine for Exercise 18 by adding $m$ to each subscript, and take the union of the two sets of five-tuples. 29. **a)** No **b)** Yes **c)** Yes **d)** Yes
31. $(s_0, B, s_1, 1, L)$, $(s_0, 1, s_1, 1, R)$, $(s_1, B, s_0, 1, R)$

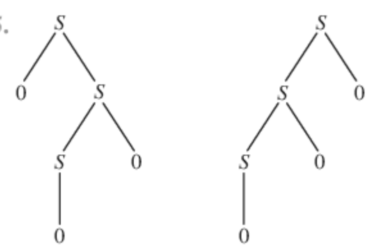
## Supplementary Exercises

1. **a)** $S \rightarrow 00S111$, $S \rightarrow \lambda$ **b)** $S \rightarrow AABS$, $AB \rightarrow BA$, $BA \rightarrow AB$, $A \rightarrow 0$, $B \rightarrow 1$, $S \rightarrow \lambda$ **c)** $S \rightarrow ET$, $T \rightarrow 0TA$, $T \rightarrow 1TB$, $T \rightarrow \lambda$, $0A \rightarrow A0$, $1A \rightarrow A1$, $0B \rightarrow B0$, $1B \rightarrow B1$, $EA \rightarrow E0$, $EB \rightarrow E1$, $E \rightarrow \lambda$

3.

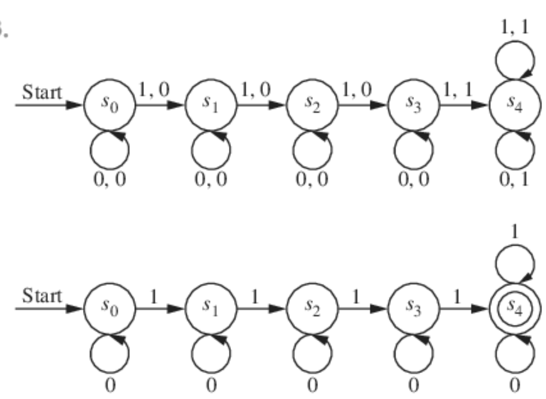


**5.**

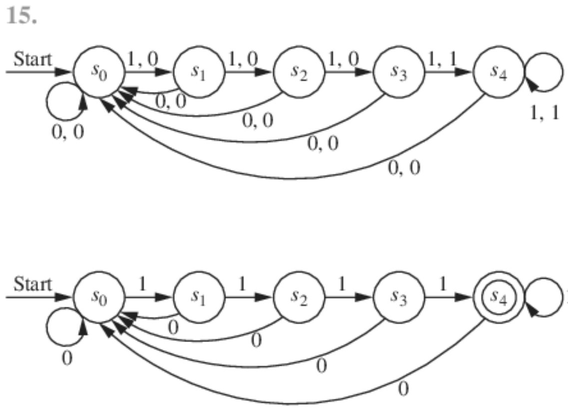


**7.** No, take $A = \{1, 10\}$ and $B = \{0, 00\}$. **9.** No, take $A = \{00, 000, 00000\}$ and $B = \{00, 000\}$. **11. a)** 1 **b)** 1 **c)** 2 **d)** 3 **e)** 2 **f)** 4

**13.**

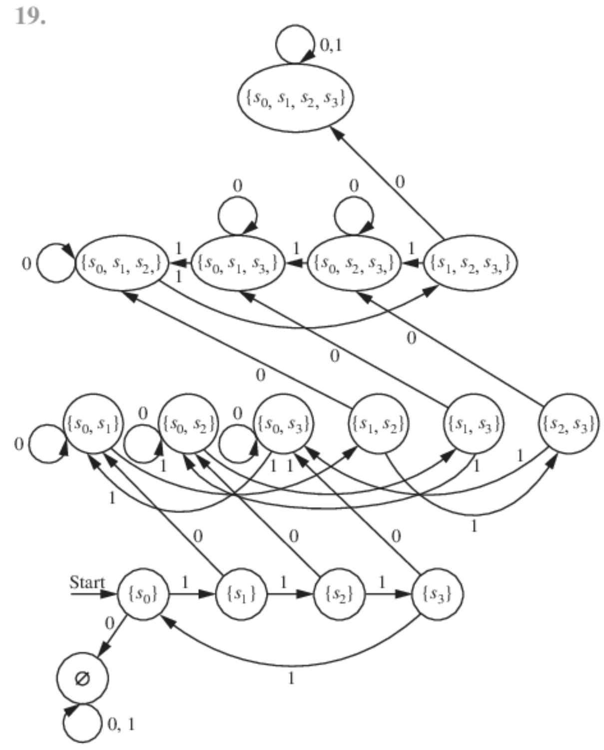


**15.**

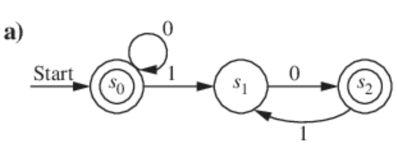


**17. a)** $n^{nk+1}m^{nk}$ **b)** $n^{nk+1}m^{n}$

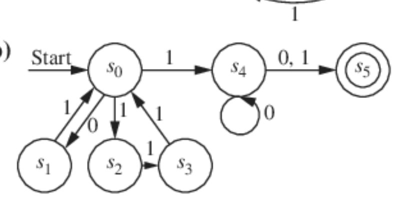
**19.**



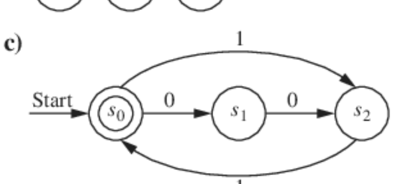
**21. a)**



**b)**

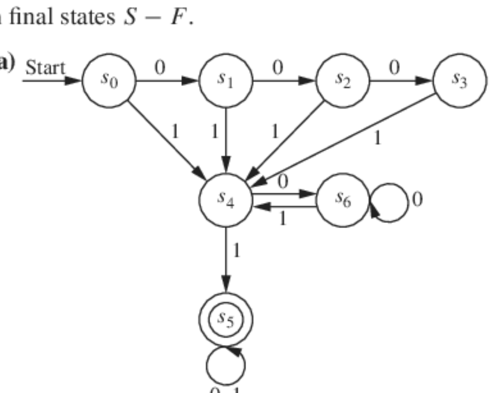


**c)**

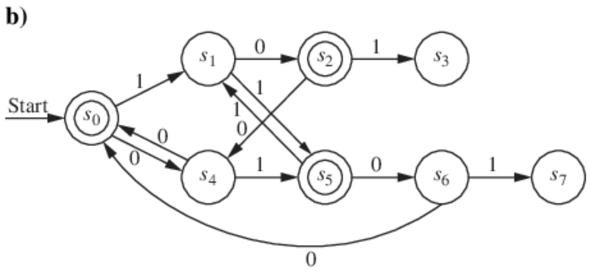


**23.** Construct the deterministic finite automaton for $A$ with states $S$ and final states $F$. For $\overline{A}$ use the same automaton but with final states $S - F$.

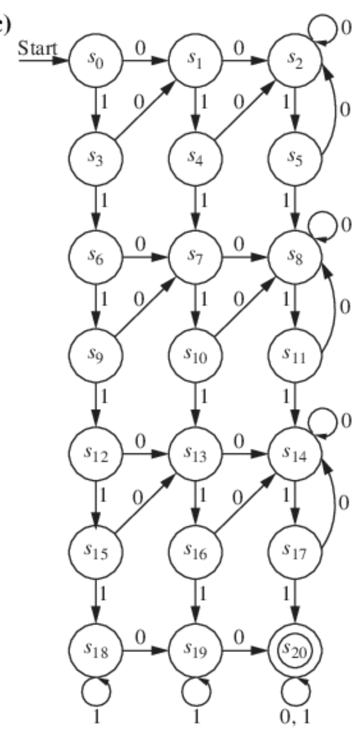
**25. a)**



**b)**




**c)**




27. Suppose that $L = \{1^p \mid p \text{ is prime}\}$ is regular, and let $S$ be the set of states in a finite-state automaton recognizing $L$. Let $z = 1^p$ where $p$ is a prime with $p > |S|$ (such a prime exists because there are infinitely many primes). By the pumping lemma it must be possible to write $z = uvw$ with $l(uv) \le |S|$, $l(v) \ge 1$, and for all nonnegative integers $i$, $uv^iw \in L$. Because $z$ is a string of all 1s, $u = 1^a$, $v = 1^b$, and $w = 1^c$, where $a+b+c = p$, $a+b \le n$, and $b \ge 1$. This means that $uv^iw = 1^a1^{bi}1^c = 1^{(a+b+c)+b(i-1)} = 1^{p+b(i-1)}$. Now take $i = p+1$. Then $uv^iw = 1^{p(1+b)}$. Because $p(1+b)$ is not prime, $uv^iw \notin L$, which is a contradiction. 29. $(s_0, *, s_5, B, L)$, $(s_0, 0, s_0, 0, R)$, $(s_0, 1, s_1, 0, R)$, $(s_1, *, s_2, *, R)$, $(s_1, 1, s_1, 1, R)$, $(s_2, 0, s_2, 0, R)$, $(s_2, 1, s_3, 0, L)$, $(s_2, B, s_4, B, L)$, $(s_3, *, s_3, *, L)$, $(s_3, 0, s_3, 0, L)$, $(s_3, 1, s_3, 1, L)$, $(s_3, B, s_0, B, R)$, $(s_4, *, s_8, B, L)$, $(s_4, 0, s_4, B, L)$, $(s_5, 0, s_5, B, L)$, $(s_5, B, s_6, B, R)$, $(s_6, 0, s_7, 1, R)$, $(s_6, B, s_6, B, R)$, $(s_7, 0, s_7, 1, R)$, $(s_7, 1, s_7, 1, R)$, $(s_8, 0, s_8, 1, L)$, $(s_8, 1, s_8, 1, L)$

# APPENDIXES

## Appendix 1

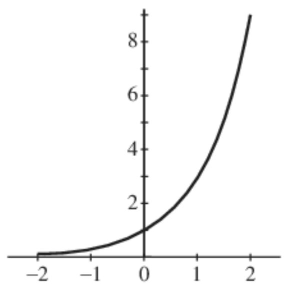
1. Suppose that $1'$ is also a multiplicative identity for the real numbers. Then, by definition, we have both $1 \cdot 1' = 1$ and $1 \cdot 1' = 1'$, so $1' = 1$. 3. For the first part, it suffices to show that $[(-x) \cdot y] + (x \cdot y) = 0$, because Theorem 2 guarantees that additive inverses are unique. Thus $[(-x) \cdot y] + (x \cdot y) = (-x + x) \cdot y$ (by the distributive law) $= 0 \cdot y$ (by the inverse law) $= y \cdot 0$ (by the commutative law) $= 0$ (by Theorem 5). The second part is almost identical. 5. It suffices to show that $[(-x) \cdot (-y)] + [-(x \cdot y)] = 0$, because Theorem 2 guarantees that additive inverses are unique: $[(-x) \cdot (-y)] + [-(x \cdot y)] = [(-x) \cdot (-y)] + [(-x) \cdot y]$ (by Exercise 3) $= (-x) \cdot [(-y) + y]$ (by the distributive law) $= (-x) \cdot 0$ (by the inverse law) $= 0$ (by Theorem 5). 7. By definition, $-(-x)$ is the additive inverse of $-x$. But $-x$ is the additive inverse of $x$, so $x$ is the additive inverse of $-x$. Therefore $-(-x) = x$ by Theorem 2. 9. It suffices to show that $(-x - y) + (x + y) = 0$, because Theorem 2 guarantees that additive inverses are unique: $(-x - y) + (x + y) = [(-x) + (-y)] + (x + y)$ (by definition of subtraction) $= [(-y) + (-x)] + (x + y)$ (by the commutative law) $= (-y) + [(-x) + (x + y)]$ (by the associative law) $= (-y) + [(-x + x) + y]$ (by the associative law) $= (-y) + (0 + y)$ (by the inverse law) $= (-y) + y$ (by the identity law) $= 0$ (by the inverse law). 11. By definition of division and uniqueness of multiplicative inverses (Theorem 4) it suffices to prove that $[(w/x) + (y/z)] \cdot (x \cdot z) = w \cdot z + x \cdot y$. But this follows after several steps, using the distributive law, the associative and commutative laws for multiplication, and the definition that division is the same as multiplication by the inverse. 13. We must show that if $x > 0$ and $y > 0$, then $x \cdot y > 0$. By the multiplicative compatibility law, the commutative law, and Theorem 5, we have $x \cdot y > 0 \cdot y = 0$. 15. First note that if $z < 0$, then $-z > 0$ (add $-z$ to both sides of the hypothesis). Now given $x > y$ and $-z > 0$, we have $x \cdot (-z) > y \cdot (-z)$ by the multiplicative compatibility law. But by Exercise 3 this is equivalent to $-(x \cdot z) > -(y \cdot z)$. Then add $x \cdot z$ and $y \cdot z$ to both sides and apply the various laws in the obvious ways to yield $x \cdot z < y \cdot z$. 17. The additive compatibility law tells us that $w + y < x + y$ and (together with the commutative law) that $x + y < x + z$. By the transitivity law, this gives the desired conclusion. 19. By Theorem 8, applied to $1/x$ in place of $x$, there is an integer $n$ (necessarily positive, because $1/x$ is positive) such that $n > 1/x$. By the multiplicative compatibility law, this means that $n \cdot x > 1$. 21. We must show that if $(w, x) \sim (w', x')$ and $(y, z) \sim (y', z')$, then $(w + y, x + z) \sim (w' + y', x' + z')$ and that $(w \cdot y + x \cdot z, x \cdot y + w \cdot z) \sim (w' \cdot y' + x' \cdot z', x' \cdot y' + w' \cdot z')$. Thus we are given that $w + x' = x + w'$ and that $y + z' = z + y'$, and we want to show that $w + y + x' + z' = x + z + w' + y'$ and that $w \cdot y + x \cdot z + x' \cdot y' + w' \cdot z' = x \cdot y + w \cdot z + w' \cdot y' + x' \cdot z'$. For the first of the desired conclusions, add the two given equations.

For the second, rewrite the given equations as $w - x = w' - x'$ and $y - z = y' - z'$, multiply them, and do the algebra.

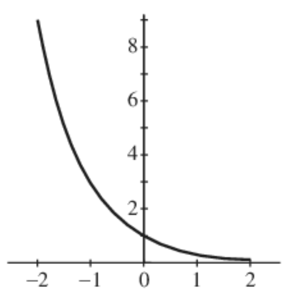
## Appendix 2

1. **a)** $2^3$ **b)** $2^6$ **c)** $2^4$ 3. **a)** $2y$ **b)** $2y/3$ **c)** $y/2$

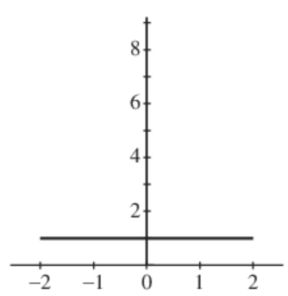
5.



**(a)**



**(b)**



**(c)**

## Appendix 3

1. After the first block is executed, $a$ has been assigned the original value of $b$ and $b$ has been assigned the original value of $c$, whereas after the second block is executed, $b$ is assigned the original value of $c$ and $a$ the original value of $c$ as well. 3. The following construction does the same thing.

$i :=$ *initial value*
**while** $i \le$ *final value*
  statement
  $i := i + 1$

# Photo Credits

**CHAPTER 1** **Page 2**: © National Library of Medicine; **p. 5**: Library of Congress, Prints and Photographs Division [LC-USZ62-61664]; **p. 11**: © University Archives. Department of Rare Books and Special Collections. Princeton University Library; **p. 20**: Courtesy Indiana University Archives; **p. 29**: By permission of the London Mathematical Society; **p. 31**: © Mary Evans Picture Library; **p. 34**: Imaging Department © President and Fellows of Harvard College: Mabel Lisle Ducasse. Henry Maurice Sheffer (1883-1964), 20th century Pastel on paper; sight: 60.9 x 46.9 cm (24 x 18 1/2 in.) Harvard Art Museums/Fogg Art Museum, Harvard University Portrait Collection, Department of Philosophy, CNA1; **p. 38**: © Bettman/Corbis; **p. 50**: © The Granger Collection, New York; **p. 97, 98**: © Master and Fellows of Trinity College, Cambridge.

**CHAPTER 2** **Page 117**: Library of Congress, Prints and Photographs Division [LC-USZ62-74393]; **p. 119**: Library of Congress, Prints and Photographs Division [LC-USZ62-49535]; **p. 120**: © The Royal Society; **p. 122**: Scientists and Inventors Portrait File Collection, Archives Center, National Museum of American History, Smithsonian Institution; **p. 163**: Courtesy Neil Sloane and AT&T Shannon Lab; **p. 171**: © AIP Emilio Segre Visual Archives, Lande Collection.

**CHAPTER 3** **Page 192**: © from THE SCIENTISTS OF THE ANCIENT WORLD, by Anderson/Stephenson. pg. 90. Illustration by Marie le Glatin Keis; **p. 207(top)**: Bachmann, Paul. *Die Arithmetik Der Quadratischen Formen*. Verlag und Druck von BG Teubner. Leipzig. Berlin. 1923; **p. 207(bottom)**: Smith Collection, Rare Book & Manuscript Library, Columbia University in the City of New York; **p. 208**: © Stanford University News Service; **p. 227**: Photo courtesy of Dr. Stephen Cook/Photo by Yvonne Heath.

**CHAPTER 4** **Page 241**: Scientists and Inventors Portrait File Collection, Archives Center, National Museum of American History, Smithsonian Institution; **p. 259**: From Math Tutor Archive; **p. 261**: © Mary Evans Picture Library; **p. 263**: © Reed Hutchinson/UCLA; **p. 267**: Scientists and Inventors Portrait File Collection, Archives Center, National Museum of American History, Smithsonian Institution; **p. 269**: © Académie des sciences-Institut de France; **p. 282**: From *A History of Science, Technology and Philosophy in the 16th and 17th Centuries*, by Abraham Wolf. Reprinted by permission of Harper Collins Publishers Ltd. © Abraham Wolf; **p. 283**: © The Mathematical Association of America 2011. All rights reserved; **p. 299**: © David Meyer, ZDNet UK; **p. 300(top)**: Courtesy Ronald L. Rivest; **p. 300(middle)**: © Kristen Tsolis; **p. 300(bottom)**: Courtesy Leonard Adleman.

**CHAPTER 5** **Page 348**: © The Granger Collection, New York; **p. 349**: © Mary Evans Picture Library; **p. 374**: Courtesy Tony Hoare; **p. 381**: © Matthew Naythons/TimePix/Getty Images.

**CHAPTER 6** **Page 400**: © The Granger Collection, New York; **p. 404**: Courtesy Mrs. Jane Burch; **p. 419**: © National Library of Medicine.

**CHAPTER 7** **Page 446**: © Roger-Viollet/The Image Works; **p. 447**: From *A Concise History of Mathematics* by Dirk Struik, 1967, Dover Publications; **p. 459**: © The Granger Collection, New York; **p. 472**: Courtesy Stephen Stigler; **p. 489**: © Académie des sciences-Institut de France; **p. 490**: Smith Collection, Rare Book & Manuscript Library, Columbia University in the City of New York.

**CHAPTER 8** **Page 507**: © KIK-IRPA, Brussels; **p. 508**: Photo courtesy of SIAM.

**CHAPTER 9** **Page 604**: Courtesy Stephen Warshall; **p. 623**: © Deutsches Museum; **p. 636**: © The Royal Society.

**CHAPTER 10** **Page 660**: © The Royal Society; **p. 695**: Scientists and Inventors Portrait File Collection, Archives Center, National Museum of American History, Smithsonian Institution; **p. 700**: © Master and Fellows of Trinity College, Cambridge; **p. 701(top)**: © Rosemarie Dirac; **p. 701(bottom)**: © Reverence to University History Photobase; **p. 706**: Julius Petersen: *Metoder og Teorier. Edition no. 6*, Det Schoenbergske Forlag, Copenhagen 1912. Photo courtesy Bjarne Toft; **p. 710**: Courtesy, Edsger W. Dijkstra; **p. 723**: Courtesy, the Archive of Polish Mathematicians, at the Institute of Mathematics of the Polish Academy of Sciences; **p. 728**: By permission of the London Mathematical Society.

**CHAPTER 11** **Page 750**: © Master and Fellows of Trinity College, Cambridge; **p. 763**: Courtesy, California State University, Santa Cruz; **p. 782**: Courtesy, the Archive of Polish Mathematicians, at the Institute of Mathematics of the Polish Academy of Sciences; **p. 798**: Courtesy, Robert Clay Prim; **p. 800**: Courtesy Joseph B. Kruskal. Photo by Rachel Kruskal.

**CHAPTER 12** **Page 812**: Courtesy MIT Museum; **p. 830**: Courtesy Maurice Karnaugh; **p. 837**: Courtesy Edward J. McCluskey; **p. 839**: Courtesy the Quine Family.

**CHAPTER 13** **Page 853**: © Donna Coveney/MIT; **p. 854(top)**: Courtesy IBM, Almaden Research Center; **p. 854(bottom)**: Courtesy Peter Naur; **p. 867**: Photo from *I Have a Photographic Memory* by Paul Halmos (1987), American Mathematical Society; **p. 872**: © Bettman/Corbis; **p. 886**: © Science Photo Library/Photo Researchers, Inc.; **p. 897**: © University Archives. Department of Rare Books and Special Collections. Princeton University Library.

**APPENDIX 1** **Page A-4**: © Hulton-Deutsch Collection/Corbis.

# Index of Biographies

# Index