Review Questions

1. a) What is a recurrence relation?
   b) Find a recurrence relation for the amount of money that will be in an account after n years if \$1,000,000 is deposited in an account yielding 9% annually.
2. Explain how the Fibonacci numbers are used to solve Fibonacci's problem about rabbits.
3. a) Find a recurrence relation for the number of steps needed to solve the Tower of Hanoi puzzle.
   b) Show how this recurrence relation can be solved using iteration.
4. a) Explain how to find a recurrence relation for the number of bit strings of length n not containing two consecutive 1s.
   b) Describe another counting problem that has a solution satisfying the same recurrence relation.
5. a) What is dynamic programming and how are recurrence relations used in algorithms that follow this paradigm?
   b) Explain how dynamic programming can be used to schedule talks in a lecture hall from a set of possible talks to maximize overall attendance.
6. Define a linear homogeneous recurrence relation of degree k.
7. a) Explain how to solve linear homogeneous recurrence relations of degree 2.
   b) Solve the recurrence relation $a_n = 13a_{n-1} - 22a_{n-2}$ for $n \geq 2$ if $a_0 = 3$ and $a_1 = 15$.
   c) Solve the recurrence relation $a_n = 14a_{n-1} - 49a_{n-2}$ for $n \geq 2$ if $a_0 = 3$ and $a_1 = 35$.
8. a) Explain how to find $f(b^k)$ where k is a positive integer if $f(n)$ satisfies the divide-and-conquer recurrence relation $f(n) = af(n/b) + g(n)$ whenever b divides the positive integer n.
   b) Find $f(256)$ if $f(n) = 3f(n/4) + 5n/4$ and $f(1) = 7$.
9. a) Derive a divide-and-conquer recurrence relation for the number of comparisons used to find a number in a list using a binary search.
   b) Give a big-O estimate for the number of comparisons used by a binary search from the divide-and-conquer recurrence relation you gave in (a) using Theorem 1 in Section 8.3.
10. a) Give a formula for the number of elements in the union of three sets.
    b) Explain why this formula is valid.
    c) Explain how to use the formula from (a) to find the number of integers not exceeding 1000 that are divisible by 6, 10, or 15.
    d) Explain how to use the formula from (a) to find the number of solutions in nonnegative integers to the equation $x_1 + x_2 + x_3 + x_4 = 22$ with $x_1 < 8$, $x_2 < 6$, and $x_3 < 5$.
11. a) Give a formula for the number of elements in the union of four sets and explain why it is valid.
    b) Suppose the sets A_1, A_2, A_3, and A_4 each contain 25 elements, the intersection of any two of these sets contains 5 elements, the intersection of any three of these sets contains 2 elements, and 1 element is in all four of the sets. How many elements are in the union of the four sets?
12. a) State the principle of inclusion–exclusion.
    b) Outline a proof of this principle.
13. Explain how the principle of inclusion–exclusion can be used to count the number of onto functions from a set with m elements to a set with n elements.
14. a) How can you count the number of ways to assign m jobs to n employees so that each employee is assigned at least one job?
    b) How many ways are there to assign seven jobs to three employees so that each employee is assigned at least one job?
15. Explain how the inclusion–exclusion principle can be used to count the number of primes not exceeding the positive integer n.
16. a) Define a derangement.
    b) Why is counting the number of ways a hatcheck person can return hats to n people, so that no one receives the correct hat, the same as counting the number of derangements of n objects?
    c) Explain how to count the number of derangements of n objects.

Review Questions

1. a) What is a relation on a set?
   b) How many relations are there on a set with n elements?
2. a) What is a reflexive relation?
   b) What is a symmetric relation?
   c) What is an antisymmetric relation?**d**
) What is a transitive relation?
3. Give an example of a relation on the set $\{1, 2, 3, 4\}$ that is
   a) reflexive, symmetric, and not transitive.
   b) not reflexive, symmetric, and transitive.
   c) reflexive, antisymmetric, and not transitive.
   d) reflexive, symmetric, and transitive.
   e) reflexive, antisymmetric, and transitive.
4. a) How many reflexive relations are there on a set with n elements?
   b) How many symmetric relations are there on a set with n elements?
   c) How many antisymmetric relations are there on a set with n elements?
5. a) Explain how an n-ary relation can be used to represent information about students at a university.
   b) How can the 5-ary relation containing names of students, their addresses, telephone numbers, majors, and grade point averages be used to form a 3-ary relation containing the names of students, their majors, and their grade point averages?
   c) How can the 4-ary relation containing names of students, their addresses, telephone numbers, and majors and the 4-ary relation containing names of students, their student numbers, majors, and numbers of credit hours be combined into a single n-ary relation?
6. a) Explain how to use a zero–one matrix to represent a relation on a finite set.
   b) Explain how to use the zero–one matrix representing a relation to determine whether the relation is reflexive, symmetric, and/or antisymmetric.
7. a) Explain how to use a directed graph to represent a relation on a finite set.
   b) Explain how to use the directed graph representing a relation to determine whether a relation is reflexive, symmetric, and/or antisymmetric.
8. a) Define the reflexive closure and the symmetric closure of a relation.
   b) How can you construct the reflexive closure of a relation?
   c) How can you construct the symmetric closure of a relation?
   d) Find the reflexive closure and the symmetric closure of the relation $\{(1, 2), (2, 3), (2, 4), (3, 1)\}$ on the set $\{1, 2, 3, 4\}$.
9. a) Define the transitive closure of a relation.
   b) Can the transitive closure of a relation be obtained by including all pairs (a, c) such that (a, b) and (b, c) belong to the relation?
   c) Describe two algorithms for finding the transitive closure of a relation.
   d) Find the transitive closure of the relation $\{(1,1), (1,3), (2,1), (2,3), (2,4), (3,2), (3,4), (4,1)\}$.
10. a) Define an equivalence relation.
    b) Which relations on the set $\{a, b, c, d\}$ are equivalence relations and contain (a, b) and (b, d)?
11. a) Show that congruence modulo m is an equivalence relation whenever m is a positive integer.
    b) Show that the relation $\{(a, b) \mid a \equiv \pm b \pmod 7\}$ is an equivalence relation on the set of integers.
12. a) What are the equivalence classes of an equivalence relation?
    b) What are the equivalence classes of the "congruent modulo 5" relation?
    c) What are the equivalence classes of the equivalence relation in Question 11(b)?
13. Explain the relationship between equivalence relations on a set and partitions of this set.
14. a) Define a partial ordering.
    b) Show that the divisibility relation on the set of positive integers is a partial order.
15. Explain how partial orderings on the sets A_1 and A_2 can be used to define a partial ordering on the set $A_1 \times A_2$.
16. a) Explain how to construct the Hasse diagram of a partial order on a finite set.
    b) Draw the Hasse diagram of the divisibility relation on the set $\{2, 3, 5, 9, 12, 15, 18\}$.
17. a) Define a maximal element of a poset and the greatest element of a poset.
    b) Give an example of a poset that has three maximal elements.
    c) Give an example of a poset with a greatest element.
18. a) Define a lattice.
    b) Give an example of a poset with five elements that is a lattice and an example of a poset with five elements that is not a lattice.
19. a) Show that every finite subset of a lattice has a greatest lower bound and a least upper bound.
    b) Show that every lattice with a finite number of elements has a least element and a greatest element.
20. a) Define a well-ordered set.
    b) Describe an algorithm for producing a totally ordered set compatible with a given partially ordered set.
    c) Explain how the algorithm from (b) can be used to order the tasks in a project if tasks are done one at a time and each task can be done only after one or more of the other tasks have been completed.

Review Questions

1. a) Define a simple graph, a multigraph, a pseudograph, a directed graph, and a directed multigraph.

b) Use an example to show how each of the types of graph in part (a) can be used in modeling. For example, explain how to model different aspects of a computer network or airline routes.

2. Give at least four examples of how graphs are used in modeling.

3. What is the relationship between the sum of the degrees of the vertices in an undirected graph and the number of edges in this graph? Explain why this relationship holds.**4**

. Why must there be an even number of vertices of odd degree in an undirected graph?

5. What is the relationship between the sum of the in-degrees and the sum of the out-degrees of the vertices in a directed graph? Explain why this relationship holds.

6. Describe the following families of graphs.

a) K_n, the complete graph on n vertices

b) $K_{m,n}$, the complete bipartite graph on m and n vertices **c**

) C_n, the cycle with n vertices

d) W_n, the wheel of size n

e) Q_n, the n-cube

7. How many vertices and how many edges are there in each of the graphs in the families in Question 6?**8**

. a) What is a bipartite graph?

b) Which of the graphs K_n, C_n, and W_n are bipartite?

c) How can you determine whether an undirected graph is bipartite?

9. a) Describe three different methods that can be used to represent a graph.

b) Draw a simple graph with at least five vertices and eight edges. Illustrate how it can be represented using the methods you described in part (a).

10. a) What does it mean for two simple graphs to be isomorphic?

b) What is meant by an invariant with respect to isomorphism for simple graphs? Give at least five examples of such invariants.

c) Give an example of two graphs that have the same numbers of vertices, edges, and degrees of vertices, but that are not isomorphic.

d) Is a set of invariants known that can be used to efficiently determine whether two simple graphs are isomorphic?

11. a) What does it mean for a graph to be connected?

b) What are the connected components of a graph?

12. a) Explain how an adjacency matrix can be used to represent a graph.

b) How can adjacency matrices be used to determine whether a function from the vertex set of a graph G to the vertex set of a graph H is an isomorphism?

c) How can the adjacency matrix of a graph be used to determine the number of paths of length r, where r is a positive integer, between two vertices of a graph?

13. a) Define an Euler circuit and an Euler path in an undirected graph.

b) Describe the famous Königsberg bridge problem and explain how to rephrase it in terms of an Euler circuit.

c) How can it be determined whether an undirected graph has an Euler path?

d) How can it be determined whether an undirected graph has an Euler circuit?

14. a) Define a Hamilton circuit in a simple graph.

b) Give some properties of a simple graph that imply that it does not have a Hamilton circuit.

15. Give examples of at least two problems that can be solved by finding the shortest path in a weighted graph.

16. a) Describe Dijkstra's algorithm for finding the shortest path in a weighted graph between two vertices.

b) Draw a weighted graph with at least 10 vertices and 20 edges. Use Dijkstra's algorithm to find the shortest path between two vertices of your choice in the graph.

17. a) What does it mean for a graph to be planar?

b) Give an example of a nonplanar graph.

18. a) What is Euler's formula for connected planar graphs?

b) How can Euler's formula for planar graphs be used to show that a simple graph is nonplanar?

19. State Kuratowski's theorem on the planarity of graphs and explain how it characterizes which graphs are planar.

20. a) Define the chromatic number of a graph.

b) What is the chromatic number of the graph K_n when n is a positive integer?

c) What is the chromatic number of the graph C_n when n is an integer greater than 2?

d) What is the chromatic number of the graph $K_{m,n}$ when m and n are positive integers?

21. State the four color theorem. Are there graphs that cannot be colored with four colors?

22. Explain how graph coloring can be used in modeling. Use at least two different examples.

Review Questions

1. **a)** Define a tree. **b)** Define a forest.
2. Can there be two different simple paths between the vertices of a tree?
3. Give at least three examples of how trees are used in modeling.
4. **a)** Define a rooted tree and the root of such a tree.
   b) Define the parent of a vertex and a child of a vertex in a rooted tree.
   c) What are an internal vertex, a leaf, and a subtree in a rooted tree?
   d) Draw a rooted tree with at least 10 vertices, where the degree of each vertex does not exceed 3. Identify the root, the parent of each vertex, the children of each vertex, the internal vertices, and the leaves.
5. **a)** How many edges does a tree with n vertices have?
   b) What do you need to know to determine the number of edges in a forest with n vertices?**6**

. **a)** Define a full m-ary tree.
   b) How many vertices does a full m-ary tree have if it has i internal vertices? How many leaves does the tree have?

7. **a)** What is the height of a rooted tree?
   b) What is a balanced tree?
   c) How many leaves can an m-ary tree of height h have?**8**

. **a)** What is a binary search tree?
   b) Describe an algorithm for constructing a binary search tree.
   c) Form a binary search tree for the words *vireo*, *warbler*, *egret*, *grosbeak*, *nuthatch*, and *kingfisher*.**9**

. **a)** What is a prefix code?
   b) How can a prefix code be represented by a binary tree?

10. **a)** Define preorder, inorder, and postorder tree traversal.
    b) Give an example of preorder, postorder, and inorder traversal of a binary tree of your choice with at least 12 vertices.
11. **a)** Explain how to use preorder, inorder, and postorder traversals to find the prefix, infix, and postfix forms of an arithmetic expression.
    b) Draw the ordered rooted tree that represents $((x - 3) + ((x/4) + (x - y) \uparrow 3))$.
    c) Find the prefix and postfix forms of the expression in part (b).
12. Show that the number of comparisons used by a sorting algorithm to sort a list of n elements is at least $\lceil \log n! \rceil$.**13**

. **a)** Describe the Huffman coding algorithm for constructing an optimal code for a set of symbols, given the frequency of these symbols.
   b) Use Huffman coding to find an optimal code for these symbols and frequencies: A: 0.2, B: 0.1, C: 0.3, D: 0.4.

14. Draw the game tree for nim if the starting position consists of two piles with one and four stones, respectively. Who wins the game if both players follow an optimal strategy?
15. **a)** What is a spanning tree of a simple graph?
    b) Which simple graphs have spanning trees?
    c) Describe at least two different applications that require that a spanning tree of a simple graph be found.
16. **a)** Describe two different algorithms for finding a spanning tree in a simple graph.
    b) Illustrate how the two algorithms you described in part (a) can be used to find the spanning tree of a simple graph, using a graph of your choice with at least eight vertices and 15 edges.
17. **a)** Explain how backtracking can be used to determine whether a simple graph can be colored using n colors.
    b) Show, with an example, how backtracking can be used to show that a graph with a chromatic number equal to 4 cannot be colored with three colors, but can be colored with four colors.
18. **a)** What is a minimum spanning tree of a connected weighted graph?
    b) Describe at least two different applications that require that a minimum spanning tree of a connected weighted graph be found.
19. **a)** Describe Kruskal's algorithm and Prim's algorithm for finding minimum spanning trees.
    b) Illustrate how Kruskal's algorithm and Prim's algorithm are used to find a minimum spanning tree, using a weighted graph with at least eight vertices and 15 edges.

12 / Boolean Algebra

Review Questions

1. Define a Boolean function of degree n.

2. How many Boolean functions of degree two are there?

3. Give a recursive definition of the set of Boolean expressions.

4. a) What is the dual of a Boolean expression?

b) What is the duality principle? How can it be used to find new identities involving Boolean expressions?

5. Explain how to construct the sum-of-products expansion of a Boolean function.

6. a) What does it mean for a set of operators to be functionally complete?

b) Is the set $\{+, \cdot\}$ functionally complete?

c) Are there sets of a single operator that are functionally complete?

7. Explain how to build a circuit for a light controlled by two switches using OR gates, AND gates, and inverters.

8. Construct a half adder using OR gates, AND gates, and inverters.

9. Is there a single type of logic gate that can be used to build all circuits that can be built using OR gates, AND gates, and inverters?

10. a) Explain how K-maps can be used to simplify sum-of-products expansions in three Boolean variables.

b) Use a K-map to simplify the sum-of-products expansion $xyz + x\bar{y}z + x\bar{y}\bar{z} + \bar{x}yz + \bar{x}\bar{y}\bar{z}$.

11. a) Explain how K-maps can be used to simplify sum-of-products expansions in four Boolean variables.

b) Use a K-map to simplify the sum-of-products expansion $wxyz + wxy\bar{z} + wx\bar{y}z + wx\bar{y}\bar{z} + w\bar{x}yz + w\bar{x}\bar{y}z + \bar{w}xyz + \bar{w}\bar{x}yz + \bar{w}\bar{x}y\bar{z}$.

12. a) What is a *don't care* condition?

b) Explain how *don't care* conditions can be used to build a circuit using OR gates, AND gates, and inverters that produces an output of 1 if a decimal digit is 6 or greater, and an output of 0 if this digit is less than 6.**13**

. a) Explain how to use the Quine–McCluskey method to simplify sum-of-products expansions.

b) Use this method to simplify $xy\bar{z} + x\bar{y}\bar{z} + \bar{x}yz + \bar{x}\bar{y}z + \bar{x}\bar{y}\bar{z}$.

13 / Modeling Computation

Review Questions

1. a) Define a phrase-structure grammar.

b) What does it mean for a string to be derivable from a string w by a phrase-structure grammar G?

2. a) What is the language generated by a phrase-structure grammar G?

b) What is the language generated by the grammar G with vocabulary $\{S, 0, 1\}$, set of terminals $T = \{0, 1\}$, starting symbol S, and productions $S \to 000S$, $S \to 1$?

c) Give a phrase-structure grammar that generates the set $\{01^n \mid n = 0, 1, 2, \ldots\}$.

3. a) Define a type 1 grammar.

b) Give an example of a grammar that is not a type 1 grammar.

c) Define a type 2 grammar.

d) Give an example of a grammar that is not a type 2 grammar but is a type 1 grammar.

e) Define a type 3 grammar.

f) Give an example of a grammar that is not a type 3 grammar but is a type 2 grammar.

4. a) Define a regular grammar.

b) Define a regular language.

c) Show that the set $\{0^m 1^n \mid m, n = 0, 1, 2, \ldots\}$ is a regular language.

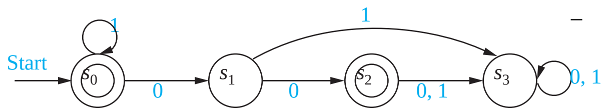
5. a) What is Backus–Naur form?

b) Give an example of the Backus–Naur form of the grammar for a subset of English of your choice.

6. a) What is a finite-state machine?

b) Show how a vending machine that accepts only quarters and dispenses a soft drink after 75 cents has been deposited can be modeled using a finite-state machine.

7. Find the set of strings recognized by the deterministic finite-state automaton shown here.



8. Construct a deterministic finite-state automaton that recognizes the set of bit strings that start with 1 and end with 1.

9. a) What is the Kleene closure of a set of strings?

b) Find the Kleene closure of the set $\{11, 0\}$.

10. a) Define a finite-state automaton.

b) What does it mean for a string to be recognized by a finite-state automaton?

11. a) Define a nondeterministic finite-state automaton.

b) Show that given a nondeterministic finite-state automaton, there is a deterministic finite-state automaton that recognizes the same language.

12. a) Define the set of regular expressions over a set I.

b) Explain how regular expressions are used to represent regular sets.

13. State Kleene's theorem.

14. Show that a set is generated by a regular grammar if and only if it is a regular set.

15. Give an example of a set not recognized by a finite-state automaton. Show that no finite-state automaton recognizes it.

16. Define a Turing machine.

17. Describe how Turing machines are used to recognize sets.

18. Describe how Turing machines are used to compute number-theoretic functions.

19. What is an unsolvable decision problem? Give an example of such a problem.