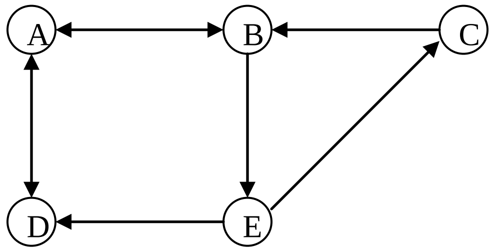
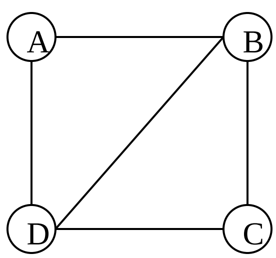
1. (a) Let a directed graph G_1 be given.



Does each of the following list of vertices form a path in G_1? If yes, determine (by circling) if the path is simple, if it is a circuit, and give its length.

a, b, e, c, b	Yes [simple circuit length ☐] No
a, d, a, d, a	Yes [simple circuit length ☐] No
a, d, e, b, a	Yes [simple circuit length ☐] No
a, b, e, c, b, a	Yes [simple circuit length ☐] No

(b) For the simple graph G_2



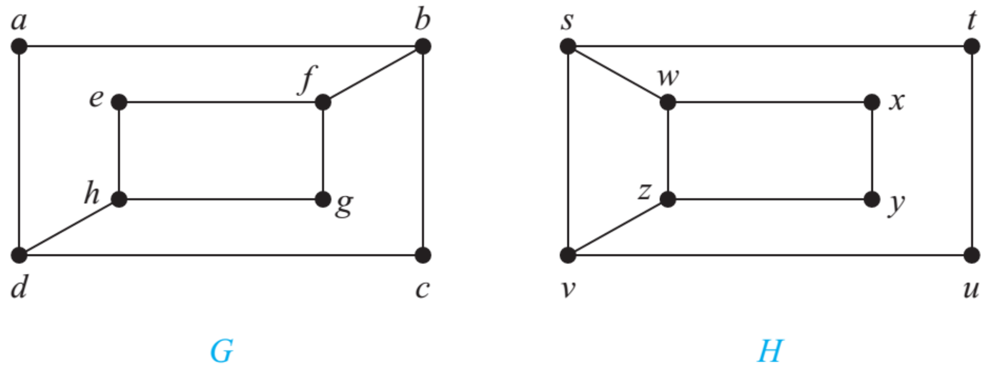
Find M^2, where M is the adjacency matrix of G_2

$M^2 =$

Find the number of paths from A to D in G_2 of length 2. ☐

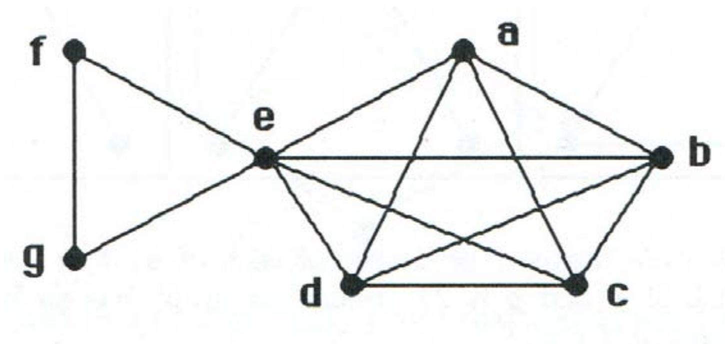
2. Let $f(n) = 1 + 2 + 3 + \ldots + n$. Show that $f(n)$ is $O(n^2)$. Be sure to specify the values of the witnesses C and k.

3. Determine whether the given pair of graphs is isomorphic. Exhibit an isomorphism or provide a rigorous argument that none exists.



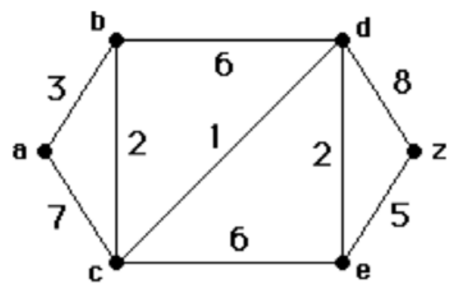
4. Show that a vertex c in the connected simple graph G is a cut vertex if and only if there are vertices u and v, both different from c, such that every path between u and v passes through c.

5.
(a) Is there an Euler circuit in the following graph? If so, find such a circuit. If not, explain why no such circuit exists.
(b) Is there a Hamilton circuit in the following graph? If so, find such a circuit. If not, prove why no such circuit exists.



6. Let $f(n) = 2n^2 + 5n\log(n) + 8n + 7$. Show that $f(n)$ is $O(n^2)$. Be sure to specify the values of the witnesses C and k.

7. Use Dijkstra's algorithm to find the length of the shortest path between the vertices *a* and *z* in the following weighted graph.



a	*b*	*c*	*d*	*e*	*z*	*S*
0	∞	∞	∞	∞	∞	*a*
X						
X						
X						
X						
X						
X						
X						
X						

8. How many vertices and how many edges does each of the following graphs have?
(a) K_5

(b) C_4

(c) W_5

(d) $K_{2,5}$

9. Describe an algorithm for finding the second largest integer in a sequence of distinct integers. Give a big-O estimate of the number of comparisons used by your algorithm.

10. Give a recursive algorithm for finding the string w^i , the concatenation of i copies of w, when w is a bit string.

11. Give a recursive definition of
a) the set of odd positive integers.
b) the set of positive integer powers of 3.
c) the set of polynomials with integer coefficients.

12. Let S be the subset of the set of ordered pairs of integers defined recursively by
Basis step: $(0, 0) \in S$.
Recursive step: If $(a, b) \in S$, then $(a + 2, b + 3) \in S$ and $(a + 3, b + 2) \in S$.
a) List the elements of S produced by the first two applications of the recursive definition.
b) Use structural induction to show that $5 \mid a + b$ when $(a, b) \in S$.

13. Write a pseudocode for an algorithm for evaluating a polynomial of degree *n*, $p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$, at $x = c$.
What is big-O estimate of the time complexity of your algorithm (in terms of the number of multiplications and additions used) as a function of *n*? Explain your answer.

14. Show that $\log(\boldsymbol{n}!)$ is $\Theta(\boldsymbol{n} \cdot \log(\boldsymbol{n}))$.

15. Prove that 1*1! + 2*2!+…+n*n! = (n + 1)! – 1 whenever n is a positive integer.

16. Prove that 1 + 3 +…+ (2n – 1) = n^2 whenever n is a positive integer.