CRASH COURSE GATE 2025

Artifical Intelligence

Uninformed and Informed Search (Part-2)

Q1 Consider the depth-first-search of an undirected graph with 3 vertices P, Q, and R. Let discovery time d(u) represent the time instant when the vertex 'u' is first visited, and finish time f(u) represent the time instant when the vertex 'u' is last visited. Given that

d(P) = 5 units	f(P) = 12 units
d(Q) = 6 units	f(Q) = 10 units
d(R) = 14 units	f(R) = 18 units

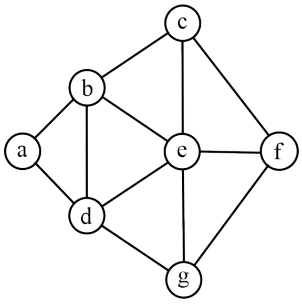
Which one of the following statements is TRUE about the graph?

(A) There is only one connected component.
(B) There are two connected components, and P and R are connected.
(C) There are two connected components, and Q and R are connected.
(D) There are two connected components, and P and Q are connected.

Q2 Consider the following sequence of nodes for the undirected graph given below:

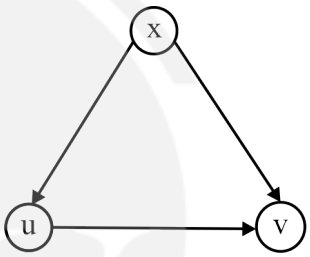
1. abefdgc
2. abefcgd
3. adgebcf
4. adbcgef

A Depth First Search (DFS) is started at node 'a'. The nodes are listed in the order they are first visited. Which of the above is/are possible output(s)?



(A) 1 and 3 only
(B) 2 and 3 only
(C) 2, 3 and 4 only
(D) 1, 2 and 3 only

Q3 A depth-first search is performed on a directed acyclic graph. Let d[u] denote the time at which vertex 'u' is visited for the first time and f[u] the time at which the DFS call to the vertex 'u' terminates. Which of the following statements is always TRUE for all edges (u, v) in the graph ?



(A) d[u] < d[v]
(B) d[u] < f[v]
(C) f[u] < f[v]
(D) f[u] > f[v]

Q4 Let G be a graph with n vertices and m edges. What is the tightest upper bound on the running time of Depth First Search on G, when G is represented as an adjacency matrix?

(A) O(n)
(B) O(n + m)
(C) $O(n^2)$
(D) $O(m^2)$

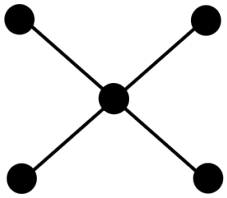
Q5 Consider the tree arcs of a BFS traversal from a source node W in an unweighted, connected, undirected graph. The tree T formed by the tree arcs is a data structure for computing

(A) The shortest path between every pair of vertices.
(B) The shortest path from W to every vertex in the graph.
(C) The shortest paths from W to only those nodes that are leaves of T.
(D) The longest path in the graph.

Q6 Consider an undirected, unweighted graph G. Let a breadth-first traversal of G be done starting from a node 'r'. Let d(r, u) and d(r, v) be the lengths of the shortest paths from 'r' to 'u' and 'v' respectively in G. If 'u' is visited before 'v' during the breadth-first traversal, which of the following statements is correct?

(A) d(r, u) < d(r, v)
(B) d(r, u) > d(r, v)
(C) d(r, u) <= d(r, v)
(D) None of the above

Q7 Consider the following undirected graph on 5 nodes.



Assume you are performing breadth-first search on this graph using a queue data structure. How many unique breadth first orderings are possible on this graph?

(A) 9 (B) 24
(C) 48 (D) 120

Q8 Let T be a depth first search tree in an undirected graph G. Vertices 'u' and 'v' are leaves of this tree T. The degrees of both 'u' and 'v' in G are at least 2.

Which one of the following statements is true?

(A) There must exist a vertex 'w' adjacent to both 'u' and 'v' in G.
(B) There must exist a vertex 'w' whose removal disconnects 'u' and 'v' in G.
(C) There must exist a cycle in G containing 'u' and 'v'.
(D) There must exist a cycle in G containing 'u' and all its neighbors in G.

Q9 Let G be an undirected graph. Consider a depth-first traversal of G, and let T be the resulting depth-first search tree. Let 'u' be a vertex in G and let 'v' be the first new (unvisited) vertex visited after visiting 'u' in the traversal. Which of the following statement is always true?

(A) {u, v} must be an edge in G, and 'u' is a descendant of 'v' in T.
(B) {u, v} must be an edge in G, and 'v' is a descendant of 'u' in T.
(C) If {u, v} is not an edge in G then 'u' is a leaf in T.
(D) If {u, v} is not an edge in G then 'u' and 'v' must have the same parent in T.

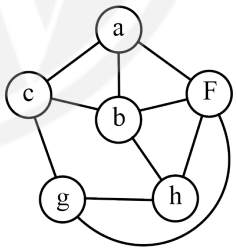
Q10 In a graph where the goal is to reach from node S to node G, the following edges and costs are given:

- S to A. cost 1
- S to B: cost 2
- A to C: cost 3
- B to C: cost 1
- C to G\ cost 2

Using Uniform Cost Search, what is the total cost of the path found by UCS if it finds the shortest path to G?

(A) 4 (B) 5
(C) 6 (D) 7

Q11 Consider the following graph:



For the graph; the following sequences of depth first search (DFS) are given

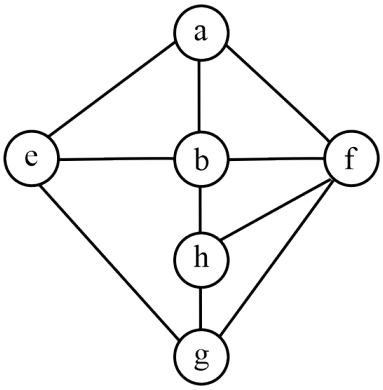
(A) abcghf
(B) abfchg
(C) abfhgc
(D) afghbc

Which of the following is correct?

(A) (A), (B) and (D) only
(B) (A), (B), (C) and (D)
(C) (B), (C) and (D) only
(D) (A), (C) and (D) only

GATE

Q12 Consider the following graph.



Among the following sequences

I. a b e g h f II. a b f e h g

III. a b f h g e IV. a f g h b e

Which are depth first traversals of the above graph?

(A) I, II, and IV only

(B) I and IV only

(C) II, III, and IV only

(D) I, III, and IV only

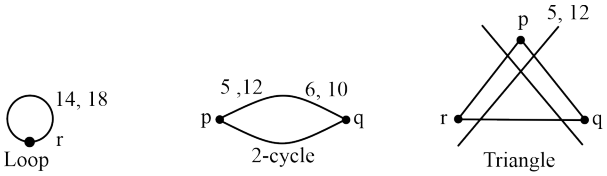
GATE

Answer Key

Q1	(D)
Q2	(B, C)
Q3	(D)
Q4	(C)
Q5	(B)
Q6	(C)
Q7	(C)
Q8	(D)
Q9	(C)
Q10	(B)
Q11	(A, C, D)
Q12	(D)

Hints & Solutions

Q1 Text Solution:



As seen in question, after 10 we have to go for 'p' again and since 'p' is finished and then 'r' is started it means 'r' must be disconnected. If there is an edge from 'q' to 'r' then 'r' must be visited before 'q' and 'p' end.

Q2 Text Solution:

1. After 'f' is visited, 'c' or 'g' should be visited next. So, the traversal is incorrect.
2. After 'c' is visited, 'e' or 'f' should be visited next. So, the traversal is incorrect.

Q3 Text Solution:

Disproving all the wrong options here:

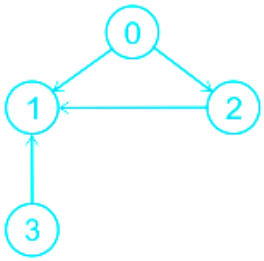
A. d[u] < d[v], Counter Example => Well if we directly start DFS on V first, then I call DFS on 'x' which visits 'u'.

B. d[u] < f[v], Counter Example => Same as A.

D. f[u] < f[v], Counter Example => Same as A again.

Q4 Text Solution:

Depth First Search of a graph takes O(m+n) time when the graph is represented using an adjacency list. In adjacency matrix representation, the graph is represented as an "n x n" matrix. To do DFS, for every vertex, we traverse the row corresponding to that vertex to find all adjacent vertices (In adjacency list representation we traverse only the adjacent vertices of the vertex). Therefore time complexity becomes $O(n^2)$



Q5 Text Solution:

BFS always has a starting node. It does not calculate the shortest path between every pair but it computes the shortest path between W and any other vertex.

Q6 Text Solution:

BFS is used to count shortest path from source (If all path costs are 1).

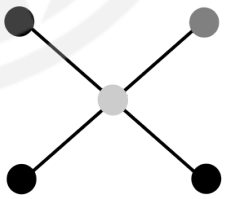
Now, if 'u' is visited before 'v' it means 2 things :-

- Either 'u' is closer to 'v', or
- If 'u' & 'v' are the same distance from 'r', then our BFS algo chose to visit 'u' before 'v'.

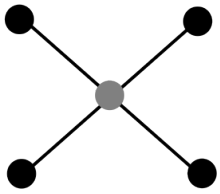
Q7 Text Solution:

Here we select the first red one from outside nodes, then select the mid one (green).

Now we have 3 choices like this for a total of 4 outside nodes we have 4.3! = 24 choices.

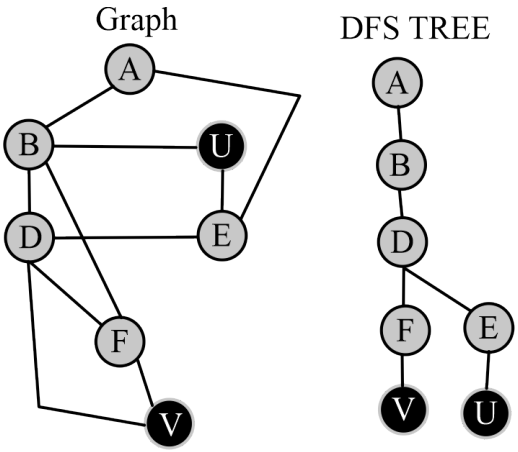


Now we will start with mid(red) one first then we have 4 choices so, total 4! = 24 choices.



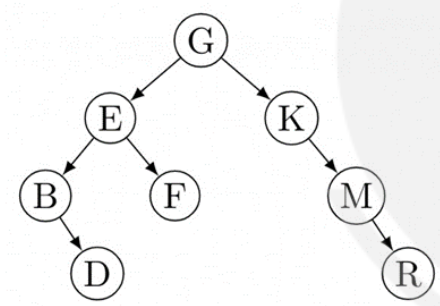
Total = 24 + 24 = 48.

Q8 Text Solution:



One diagram, which is eliminating option A, B, C.
Hence D is the answer.
A leaf in a DFS tree has degree two.
Another leaf in a DFS tree has degree two.
Both can be non adjacent leaves also.
It implies that if a leaf has degree two then it will form a cycle.
Since it is non adjacent leaves that have cycle, Option C may be wrong. Hence option D.

Q9 Text Solution:



Let this be the DFS order of the tree, then,
u = D & v = F.
So, we conclude that

1. It is not necessary that there is an edge between them.
2. If there is no edge then ‘u’ must be leaf i.e. ‘D’ is leaf here.
3. It is not always possible that ‘u’ and ‘v’ have same parent. But they have the same ancestor.

Q10 Text Solution:

1. **Initial State:**
   - Start at S.
   - Move to A with cost 1 or B with cost 2.
2. **Expand Node A:**
   - From A, move to C with an additional cost of 3. Total cost to C is 1 + 3 = 4.
   - From S to A to C, the path cost is 4.
3. **Expand Node B:**
   - From B, move to C with an additional cost of 1. Total cost to C is 2 + 1 = 3. This path to C is cheaper than via A
4. **Expand Node C:**
   - From C, move to G with an additional cost of 2. Total cost to G is 3 4- 2 = 5.

The total cost of the shortest path from S to G found by UCS is 5.

Q11 Text Solution:

The depth-first search (DFS) algorithm visits all the vertices of a graph by exploring as far as possible along each branch before backtracking. In a graph with multiple connected components, the DFS will be performed for each component separately.
In the given graph, the DFS order can be different depending on the starting vertex and the order in which the edges are traversed.
(A) abcghf is a valid DFS order, starting at vertex "a".
(C) abfhgc is also a valid DFS order, starting at vertex "a".
(D) afghbc is a valid DFS order, starting at vertex "a".
However, (B) abfchg is not a valid DFS order, as the edge from "g" to "f" is not visited.
Therefore, the correct answer is (A), (C) and (D) only.

Q12 Text Solution:

Depth First Search (DFS) is graph traversal algorithm uses stack data structure.
Option I: a b e g h f : visit a; insert connected node of a, b; insert connected node of b, e ; insert connected node of e, g ; insert connected node of g, h ; insert connected node of h,f.
Option II : a b f e h g : visit a; insert connected node of a, b; insert connected node of b, f ; Next

to insert the connected node of f, which is either g or h; but e is not possible.
Option III: a b f h g e : visit a; insert connected node of a, b; insert connected node of b, f ; insert connected node of f, h ; insert connected node of h, g ; insert connected node 01 g,e.

Option IV : a f g h b e: visit a; insert connected node of a, f; insert connected node of f, g ; insert connected node of g, h ; insert connected node of h, b ; insert connected node of b, e.