MODULE 1: INTRODUCTION TO GRAPH THEORY

Graph

A **graph** G=(V,E)G = (V, E)G=(V,E) consists of:

- **V** → set of **vertices (nodes)**
- **E** → set of **edges (links)** connecting vertices

Types:

- **Simple graph** – no loops, no multiple edges
- **Multigraph** – multiple edges allowed
- **Pseudograph** – loops allowed

Basic Terminology

- **Degree of a vertex**: Number of edges incident to it
- **Pendant vertex**: Vertex of degree 1
- **Isolated vertex**: Vertex of degree 0
- **Loop**: Edge connecting a vertex to itself
- **Adjacent vertices**: Vertices connected by an edge

Walk, Path, Circuit

- **Walk**: Sequence of vertices where edges may repeat
- **Trail**: Walk with no repeated edges
- **Path**: Walk with no repeated vertices
- **Circuit (Cycle)**: Closed path (start = end)

📌 *Exam tip*: Path → no repetition, Walk → repetition allowed

---

Connectedness

- A graph is **connected** if there is a path between **every pair of vertices**
- If not connected → **disconnected graph**
- Each connected part is called a **component**

Handshaking Lemma

Statement:

Sum of degrees of all vertices = 2×2 \times2× number of edges

∑deg (v)=2∣E∣\sum \deg(v) = 2|E|∑deg(v)=2∣E∣

Important result:

- Number of vertices with **odd degree is always even**

📌 Very important for short questions

Isomorphism

Two graphs are **isomorphic** if:

- They have the same structure
- Only vertex labels differ

To check:

- Same number of vertices
- Same number of edges
- Same degree sequence

Subgraph

A graph formed from a **subset of vertices and edges** of a graph

Types:

- **Proper subgraph**
- **Induced subgraph**

Reachability

- Vertex **v** is reachable from **u** if there exists a path from **u to v**

Union and Intersection of Graphs

- **Union**: Combines vertices and edges of both graphs
- **Intersection**: Common vertices and edges only

MODULE 2: GRAPH TRANSVERSAL

Euler Graph

A graph is **Eulerian** if:

- It is connected
- **All vertices have even degree**

Euler Path

- Exactly **two vertices** have odd degree

📌 Euler graph → covers every edge **exactly once**

Shortest Path Problem

Finding the **minimum distance path** between two vertices

Common algorithm:

- **Dijkstra's Algorithm**

Applications:

- GPS navigation
- Network routing

Hamiltonian Graph

- Contains a **Hamiltonian cycle**
- Visits **every vertex exactly once**

📌 No simple test like Euler graphs
📌 Often solved by logic or conditions

Traveling Salesman Problem (TSP)

- Find the **shortest possible Hamiltonian cycle**
- NP-hard problem

Applications:

- Delivery routing
- Scheduling

Bipartite Graph

Vertices divided into **two disjoint sets**

- Edges connect only between sets, not within

Key property:

- **No odd-length cycles**

Examples:

- Matching problems
- Job assignment

MODULE 3: TREES

Tree

A **tree** is a graph that is:

- Connected
- Acyclic (no cycles)

Properties:

- For **n vertices**, edges = **n − 1**
- Exactly **one path** between any two vertices

Rooted Tree

- One vertex chosen as **root**
- Levels defined from root

Terms:

- Parent
- Child
- Leaf (degree 1)

Path Length in Rooted Trees

- Number of edges between root and a vertex
- Used to measure **depth**

Spanning Tree

- A subgraph that:
  - Includes all vertices
  - Is a tree

📌 Graph can have **many spanning trees**

Fundamental Circuit

- Formed by adding one edge to a spanning tree
- Exactly **one cycle** is created

Weighted Graph

Edges have weights (cost, distance, time)

Cut Set and Cut Vertex

- **Cut vertex**: Removing it disconnects the graph
- **Cut set**: Set of edges whose removal disconnects the graph

---

Fundamental Cut Set

- Cut set associated with a particular spanning tree edge

Minimum Spanning Tree (MST)

Spanning tree with **minimum total weight**

Algorithms:

- **Kruskal's Algorithm**
- **Prim's Algorithm**

Applications:

- Network design
- Wiring systems

MODULE 4: DIRECTED & PLANAR GRAPHS

Directed Graph (Digraph)

Edges have **direction**

Terms:

- **In-degree**
- **Out-degree**

Directed Connectedness

- **Strongly connected**: Path exists in both directions
- **Weakly connected**: Connected ignoring direction

Directed Trees

- Exactly one root
- All edges directed away from or toward root

Network Flow

Graph with:

- Source
- Sink
- Capacity on edges

Max Flow – Min Cut Theorem

Maximum flow from source to sink = Minimum cut capacity

📌 Very important theorem

Matrix Representation of Graph

1. **Adjacency Matrix**
2. **Incidence Matrix**

Used for:

- Computer representation
- Algorithm implementation

Planar Graph

Can be drawn in a plane **without edge crossings**

Dual Graph

- **Geometric dual**: Formed using faces
- **Combinational dual**: Based on structure

Kuratowski’s Theorem

A graph is **non-planar** if it contains:

- K5K_5K5 or
- K3,3K_{3,3}K3,3 as a subdivision

📌 High exam weight

Thickness and Crossing Number

- **Thickness**: Minimum number of planar graphs needed
- **Crossing number**: Minimum edge crossings

MODULE 5: COMBINATORICS

Partition

Ways of writing a number as a sum of positive integers

Example:

- 5 = 4+1 = 3+2 = 2+2+1

Counting Functions

Number of functions between two sets

Types:

- One-to-one (injective)
- Onto (surjective)
- Bijective

Partitions into Odd or Unequal Parts

Important result:

Number of partitions into odd parts = number of partitions into distinct parts

---

Necklaces

Arrangement of beads considering **rotations**

Used with:

- **Burnside’s Lemma**
- **Polya’s Theorem**

Euler’s Totient Function ϕ(n)\phi(n)ϕ(n)

Number of integers ≤ n that are **coprime** to n

Example:

- ϕ(8)=4\phi(8) = 4ϕ(8)=4

Set of Symmetries

Includes:

- Rotations
- Reflections

Used for:

- Enumeration problems

Enumeration (Odd & Even Cases)

Counting distinct arrangements considering symmetry

MODULE 1: INTRODUCTION TO GRAPH THEORY

Q1. Define a graph and explain its types

Answer:
A **graph** G=(V,E)G = (V, E)G=(V,E) consists of a non-empty set of vertices VVV and a set of edges EEE connecting pairs of vertices.

Types of graphs:

1. **Simple graph** – no loops or parallel edges
2. **Multigraph** – parallel edges allowed
3. **Pseudograph** – loops allowed
4. **Directed graph** – edges have direction
5. **Weighted graph** – edges have weights

Q2. Define walk, path, trail, and circuit

Answer:

- **Walk**: A sequence of vertices where edges may repeat
- **Trail**: Walk with no repeated edges
- **Path**: Walk with no repeated vertices
- **Circuit**: Closed path (start = end)

Q3. What is a connected graph?

Answer:
A graph is **connected** if there exists a path between **every pair of vertices**.
If not, it is called **disconnected**, and each connected part is called a **component**.

Q4. State and prove Handshaking Lemma

Answer:
Statement:

In any undirected graph, the sum of degrees of all vertices equals twice the number of edges.

∑deg (v)=2∣E∣\sum \deg(v) = 2|E|∑deg(v)=2∣E∣

Proof:
Each edge contributes **2 degrees** (one at each end).
Hence total degree count = 2×2 \times2× number of edges.

Result:
Number of vertices with odd degree is always even.

Q5. What is graph isomorphism?

Answer:
Two graphs are **isomorphic** if they have the same structure but different labels.

Conditions:

- Same number of vertices
- Same number of edges
- Same degree sequence

MODULE 2: GRAPH TRANSVERSAL

Q6. Define Euler graph and Euler path

Answer:

- **Euler graph**: A connected graph in which **all vertices have even degree**.
- **Euler path**: A path that uses **every edge exactly once**.

Q7. State Euler's theorem

Answer:

A connected graph is Eulerian **if and only if** every vertex has even degree.

Q8. Define Hamiltonian graph

Answer:
A graph is **Hamiltonian** if it contains a **Hamiltonian cycle**, i.e., a cycle that visits **every vertex exactly once**.

Q9. Explain Traveling Salesman Problem (TSP)

Answer:
TSP finds the **shortest Hamiltonian cycle** visiting every vertex exactly once and returning to the starting point.
It is an **NP-hard problem**.

Q10. Define bipartite graph

Answer:
A graph is **bipartite** if its vertices can be divided into two disjoint sets such that edges connect only vertices from different sets.

Property:
Bipartite graphs contain **no odd cycles**.

MODULE 3: TREES

Q11. Define a tree and state its properties

Answer:
A **tree** is a connected graph with no cycles.

Properties:

- If a tree has nnn vertices, it has n−1n-1n−1 edges
- Exactly one path exists between any two vertices
- Removing any edge disconnects the tree

Q12. Prove that a tree with n vertices has n−1 edges

Answer:
A tree is minimally connected.
If it had fewer than n−1n-1n−1 edges → disconnected
If more than n−1n-1n−1 edges → cycle formed
Therefore, a tree has exactly n−1n-1n−1 edges.

Q13. What is a rooted tree?

Answer:
A **rooted tree** is a tree in which one vertex is designated as the **root**.
Edges are oriented away from or toward the root.

Q14. Define spanning tree

Answer:
A **spanning tree** is a subgraph that:

- Includes all vertices
- Is connected
- Contains no cycles

Q15. What is a minimum spanning tree?

Answer:
A **minimum spanning tree (MST)** is a spanning tree of a weighted graph with **minimum total edge weight**.

Algorithms:

- Kruskal's Algorithm
- Prim's Algorithm

MODULE 4: DIRECTED & PLANAR GRAPHS

Q16. Define directed graph

Answer:
A **directed graph (digraph)** has edges with direction.

- **In-degree**: Number of incoming edges
- **Out-degree**: Number of outgoing edges

Q17. Explain strong and weak connectedness

Answer:

- **Strongly connected**: Path exists in both directions between every pair
- **Weakly connected**: Connected if direction is ignored

Q18. State Max-Flow Min-Cut theorem

Answer:

The maximum flow from source to sink equals the minimum cut capacity of the network.

Q19. What is a planar graph?

Answer:
A graph is **planar** if it can be drawn in a plane **without edge crossings**.

Q20. State Kuratowski's theorem

Answer:

A graph is planar if and only if it does not contain a subdivision of K5K_5K5 or K3,3K_{3,3}K3,3.

Q21. State Euler's formula for planar graphs

Answer:
For a connected planar graph:

V−E+F=2V - E + F = 2V−E+F=2

MODULE 5: COMBINATORICS

Q22. Define partition of a number

Answer:
A **partition** of a number is a way of writing it as a sum of positive integers, order ignored.

Example:
5 = 3 + 2 = 2 + 2 + 1

Q23. Define Euler's Totient function

Answer:
ϕ(n)\phi(n)ϕ(n) is the number of integers ≤ nnn that are **coprime to n**.

Example:
ϕ(10)=4\phi(10) = 4ϕ(10)=4 → {1, 3, 7, 9}

Q24. State partition theorem

Answer:

The number of partitions of a number into odd parts equals the number of partitions into distinct parts.

Q25. State Burnside's Lemma

Answer:

The number of distinct arrangements equals the average number of arrangements fixed by group symmetries.

MODULE 1: INTRODUCTION TO GRAPH THEORY

1. Simple Graph

```
A ----- B
|       |
|       |
D ----- C
```

Caption: A simple graph has no loops and no multiple edges.

---

2. Walk

```
A -- B -- C
     |    |
     ------
```

Caption: Walk allows repetition of vertices and edges.

---

3. Path

```
A -- B -- C -- D
```

Caption: Path has no repeated vertices.

---

4. Circuit (Cycle)

```
A -- B
|    |
D -- C
```

Caption: A circuit is a closed path.

---

5. Connected Graph

```
A -- B -- C
|         |
D --------
```

Caption: A graph is connected if every pair of vertices has a path.

---

6. Disconnected Graph

```
A -- B     C -- D
```

Caption: A disconnected graph has more than one component.

---

7. Degree of a Vertex

```
     B
     |
A ---C--- D
     |
     E
```

Caption: Degree of vertex C is 4.

---

8. Subgraph

```
Original:      Subgraph:
A -- B -- C    A -- B
|         |
D --------
```

Caption: A subgraph is formed from a subset of vertices and edges.

---

9. Isomorphic Graphs

```
A -- B          1 -- 2
|    |    ≅     |    |
D -- C          4 -- 3
```

Caption: Isomorphic graphs have the same structure but different labels.

MODULE 2: GRAPH TRANSVERSAL

10. Euler Graph

```
A -- B
|    |
D -- C
```

Caption: Euler graph has all vertices of even degree.

---

11. Euler Path

```
A -- B -- C -- D
```

Caption: Euler path exists when exactly two vertices have odd degree.

---

12. Hamiltonian Cycle

```
A -- B
|    |
D -- C
```

Caption: Hamiltonian cycle visits every vertex exactly once.

---

13. Bipartite Graph

```
U-set        V-set
 A ----- 1
 B ----- 2
 C ----- 3
```

Caption: Vertices can be divided into two sets with no edges within a set.

---

14. Odd Cycle (Not Bipartite)

```
A -- B
 \  /
  C
```

Caption: Presence of odd cycle means graph is not bipartite.

MODULE 3: TREES

15. Tree

```
    A
   / \
  B   C
     / \
    D   E
```

Caption: A tree is connected and has no cycles.

---

16. Rooted Tree

```
        A (Root)
       / \
      B   C
         / \
        D   E
```

Caption: Rooted tree has a designated root vertex.

---

17. Unique Path in Tree

```
A -- B -- C -- D
```

Caption: Exactly one path exists between any two vertices in a tree.

---

18. Spanning Tree

```
Graph:           Spanning Tree:
A -- B           A -- B
| \/ |           |
| /\ |           D -- C
D -- C
```

Caption: A spanning tree includes all vertices with no cycles.

---

19. Cut Vertex

```
A -- B -- C
        |
        D
```

Caption: Removing B disconnects the graph.

---

20. Minimum Spanning Tree

```
A --2-- B
|       |
3       1
|       |
```

```
D --4-- C
```

Caption: MST has minimum total edge weight.

MODULE 4: DIRECTED & PLANAR GRAPHS

21. Directed Graph

```
A ---> B ---> C
^             |
|-------------|
```

Caption: Edges have direction in a directed graph.

---

22. Strongly Connected Digraph

```
A ---> B
^      |
|      v
D <--- C
```

Caption: Path exists in both directions between every pair.

---

23. Network Flow

```
(S) ---> A ---> (T)
  \        |
   ---> B ---
```

Caption: Flow moves from source to sink with capacities.

---

24. Planar Graph

```
A -- B
|    |
D -- C
```

Caption: Can be drawn without edge crossings.

---

25. Non-Planar Graph ($K_{3,3}$)

```
A   B   C
|\  |\  |
| \ | \ |
1  2  3
```

Caption: $K_{3,3}$ is non-planar.

MODULE 5: COMBINATORICS

26. Partition of a Number

```
5 = 3 + 2
5 = 2 + 2 + 1
```

Caption: Partition writes a number as sum of positive integers.

---

27. Necklace (Rotation Symmetry)

```
  ● --- ●
 /       \
●         ●
 \       /
  ● --- ●
```

Caption: Rotations give identical arrangements.

---

28. Symmetry (Reflection + Rotation)

```
   ▲
  / \
 ●   ●
```

Caption: Symmetry operations reduce distinct arrangements.

---

29. Euler’s Totient Example

```
n = 10
Coprime: 1, 3, 7, 9
```

Caption: φ(n) counts integers coprime to n.

📃 GRAPH THEORY & COMBINATORICS

50 IMPORTANT MCQs

Module 1: Basics of Graph Theory

1. A graph with loops allowed but no parallel edges is called
A. Simple graph
B. Multigraph
C. Pseudograph
D. Directed graph

2. The degree of an isolated vertex is
A. 1
B. 0
C. −1
D. 2

3. In an undirected graph, each edge contributes how many degrees?
A. 1
B. 2
C. 0
D. Depends on graph

4. A walk in which edges are not repeated is called
A. Walk
B. Path
C. Trail
D. Circuit

5. A closed walk with no repeated edges is called
A. Path
B. Circuit
C. Trail
D. Euler graph

6. A graph is connected if
A. It has no isolated vertices
B. There exists a path between every pair of vertices
C. All vertices have equal degree
D. It has cycles

7. The Handshaking Lemma applies to
A. Directed graphs

B. Undirected graphs
C. Trees only
D. Weighted graphs only

8. Number of odd degree vertices in a graph is always
A. Odd
B. Even
C. Prime
D. Zero

9. Two graphs are isomorphic if they have
A. Same labels
B. Same drawing
C. Same structure
D. Same number of faces

10. A subgraph contains
A. All vertices and edges
B. Only edges
C. Subset of vertices and edges
D. Only vertices

Module 2: Euler, Hamiltonian & Bipartite Graphs

11. A connected graph is Eulerian if
A. Exactly two vertices have odd degree
B. All vertices have even degree
C. All vertices have odd degree
D. It has no cycles

12. An Euler path exists if
A. All vertices are even degree
B. Exactly two vertices are odd degree
C. More than two vertices are odd degree
D. Graph is complete

13. Which graph problem visits each edge exactly once?
A. Hamiltonian
B. Shortest path
C. Euler
D. Spanning tree

14. A Hamiltonian cycle visits
A. Every edge once
B. Every vertex once
C. Some vertices
D. Only even degree vertices

15. Traveling Salesman Problem is based on
A. Euler path
B. Bipartite matching
C. Hamiltonian cycle
D. Minimum cut

16. A bipartite graph has vertex set divided into
A. Three sets
B. Two disjoint sets
C. Even and odd sets
D. Complete sets

17. A graph is bipartite if it contains
A. Even cycles only
B. Odd cycles only
C. No cycles
D. Complete graph

18. Which of the following graphs is always bipartite?
A. Cycle of length 3
B. Cycle of length 5
C. Cycle of length 4
D. Complete graph K_5

19. Dijkstra's algorithm is used to find
A. Euler path
B. Minimum spanning tree
C. Shortest path
D. Hamiltonian cycle

20. Which problem is NP-hard?
A. Euler path
B. TSP
C. BFS
D. Spanning tree

Module 3: Trees & Spanning Trees

21. A tree is a graph which is
A. Cyclic and connected
B. Acyclic and connected
C. Disconnected
D. Complete

22. A tree with n vertices has how many edges?
A. n
B. n + 1

C. n − 1
D. 2n

23. In a tree, removing any edge will make the graph
A. Cyclic
B. Complete
C. Disconnected
D. Directed

24. A vertex of degree 1 in a tree is called
A. Root
B. Leaf
C. Cut vertex
D. Parent

25. A spanning tree of a graph
A. Has cycles
B. Includes all vertices
C. Has maximum edges
D. Is always unique

26. A graph can have how many spanning trees?
A. Zero
B. One only
C. Exactly two
D. More than one

27. Fundamental circuit is formed by
A. Removing an edge
B. Adding one edge to spanning tree
C. Removing a vertex
D. Adding a vertex

28. A cut vertex is a vertex whose removal
A. Reduces degree
B. Adds cycles
C. Disconnects the graph
D. Makes graph complete

29. Minimum spanning tree minimizes
A. Number of vertices
B. Number of edges
C. Total edge weight
D. Degree of vertices

30. Prim's algorithm is used for
A. Shortest path
B. MST
C. Euler circuit
D. Hamiltonian cycle

Module 4: Directed & Planar Graphs

31. In-degree of a vertex is
A. Total edges
B. Outgoing edges
C. Incoming edges
D. Undirected edges

32. A digraph is strongly connected if
A. Direction is ignored
B. Path exists in one direction
C. Path exists in both directions
D. All degrees are equal

33. Network flow graph must have
A. Only source
B. Only sink
C. Source and sink
D. No vertices

34. Max-Flow Min-Cut theorem states that
A. Max flow < Min cut
B. Max flow > Min cut
C. Max flow = Min cut
D. No relation

35. Adjacency matrix of a graph with n vertices is of order
A. n × n
B. n × e
C. e × e
D. v × e

36. A planar graph can be drawn
A. With minimum edges
B. With crossings
C. Without edge crossings
D. Only using straight lines

37. Euler's formula for planar graph is
A. V + E + F = 2
B. V − E + F = 2
C. V × E = F
D. E − V = 2

38. Which graph is non-planar?
A. K_4
B. K_3

C. K_5
D. Tree

39. Kuratowski's theorem involves
A. Trees
B. Euler graphs
C. K_5 and $K_{3,3}$
D. Bipartite graphs

40. Crossing number of a planar graph is
A. 1
B. 2
C. 0
D. Infinite

Module 5: Combinatorics

41. A partition of a number ignores
A. Value
B. Order
C. Size
D. Frequency

42. Number of partitions into odd parts equals partitions into
A. Prime parts
B. Even parts
C. Distinct parts
D. Equal parts

43. A one-to-one function is also called
A. Surjective
B. Injective
C. Bijective
D. Constant

44. A bijective function is
A. One-to-one only
B. Onto only
C. Both one-to-one and onto
D. Neither

45. Euler's Totient function φ(n) counts
A. Divisors
B. Prime numbers
C. Coprime numbers
D. Factors

46. φ(7) is
A. 7
B. 6
C. 5
D. 4

47. Necklace problems use
A. Dijkstra's algorithm
B. Polya's theorem
C. Kruskal's algorithm
D. BFS

48. Burnside's lemma is used for
A. Sorting
B. Searching
C. Enumeration
D. Optimization

49. Symmetry group of a square has how many elements?
A. 2
B. 4
C. 6
D. 8

50. Enumeration problems consider
A. Order only
B. Size only
C. Symmetry
D. Graph weight

✓ ANSWER KEY

1. C
2. B
3. B
4. C
5. B
6. B
7. B
8. B
9. C
10. C
11. B
12. B
13. C
14. B
15. C

16. B
17. A
18. C
19. C
20. B
21. B
22. C
23. C
24. B
25. B
26. D
27. B
28. C
29. C
30. B
31. C
32. C
33. C
34. C
35. A
36. C
37. B
38. C
39. C
40. C
41. B
42. C
43. B
44. C
45. C
46. B
47. B
48. C
49. D
50. C