IB DP MATHS AI · TOPIC 3 GEOMETRY AND TRIGONOMETRY · AHL 3.14–3.16 · ~8 MIN READ

Graph theory

HL graph theory for Maths AI: vertices, edges and degrees, simple, complete, weighted and directed graphs, Eulerian and Hamiltonian paths, adjacency matrices and counting walks, minimum spanning trees with Kruskal and Prim, the Chinese postman problem and the travelling salesman problem with upper and lower bounds.

Vertices, edges & degree | Simple, complete, weighted & directed graphs | Trees | Eulerian trails & circuits | Hamiltonian paths & cycles | Adjacency matrix | Counting walks | Kruskal & Prim

RS by Riyaz Sir
mentorriyaz.in

No tag = **SL and HL** HL ONLY HL students only FORMULA BOOKLET given in the exam

M1 A1 how marks are awarded highlight = comes up again and again Test yourself at the end

IN 30 SECONDS

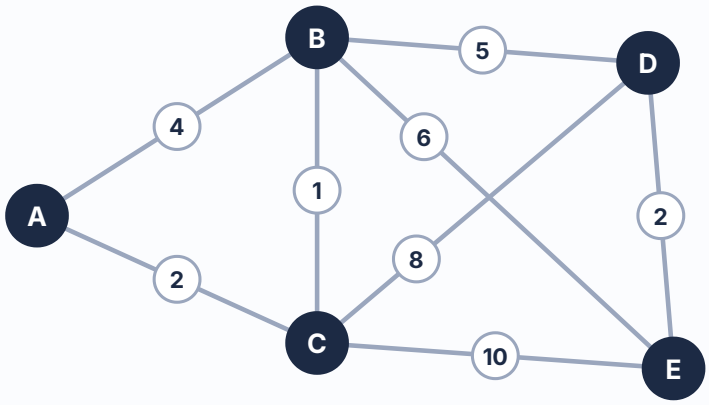
- A **graph** is a set of vertices joined by edges. The **degree** of a vertex is the number of edges at it, and the degrees always add up to **twice the number of edges**.
- **Eulerian:** uses every **edge** once. A circuit exists if every degree is even; a trail (start ≠ end) if exactly two degrees are odd. **Hamiltonian:** visits every **vertex** once.
- The **adjacency matrix** A records edges; the entries of A^k count the walks of length k.
- **Minimum spanning tree:** Kruskal (take the cheapest edge that doesn't make a cycle) or Prim (grow the tree from one vertex). **Chinese postman:** traverse every edge and return, repeating the cheapest pairing of odd vertices. **Travelling salesman:** visit every vertex and return, found between a lower and an upper bound.
- This note is HL only and unchanged for 2029.

BY THE END YOU CAN

- Use the vocabulary of graphs: vertices, edges, degree, simple, complete, weighted, directed, connected, trees and cycles. HL
- Decide whether a graph has an Eulerian trail or circuit, and recognise Hamiltonian paths and cycles. HL
- Write adjacency matrices and use powers of them to count walks. HL
- Find minimum spanning trees with Kruskal's and Prim's algorithms. HL
- Solve Chinese postman problems. HL
- Find upper and lower bounds for the travelling salesman problem. HL

Riyaz Sir

1 Graph vocabulary HL



degrees
A 2, B 4
C 4, D 3
E 3
odd: D, E

A weighted graph: 5 vertices, 8 edges, total weight 38.

A weighted graph. Numbers on the edges are weights: distance, time or cost.

Word	Meaning
simple graph	no loops and no multiple edges between the same pair of vertices
complete graph K_n	every vertex joined to every other: $\frac{n(n-1)}{2}$ edges
connected	there is a path between every pair of vertices
tree	connected with no cycles; a tree on n vertices has n – 1 edges
weighted graph	each edge has a number (weight)
directed graph	edges have directions (arrows), like one-way streets
walk / trail / path	a route; a trail repeats no edge; a path repeats no vertex
cycle	a path that returns to its start

WORKED EXAMPLE

Find the degrees of the vertices in the graph above and check the handshaking lemma.

1. A 2, B 4, C 4, D 3, E 3 A1
2. **sum = 16 = 2 × 8 edges ✓** R1

- **Eulerian?** D and E have odd degree, so there is an Eulerian **trail** from D to E (or E to D) but no Eulerian circuit.

2 Adjacency matrices and walks HL

	A	B	C	D	E
A	0	1	1	0	0
B	1	0	1	1	1
C	1	1	0	1	1
D	0	1	1	0	1
E	0	1	1	1	0

- **Row A** shows A is joined to B and C. An undirected graph has a **symmetric** matrix; the row sums are the degrees.

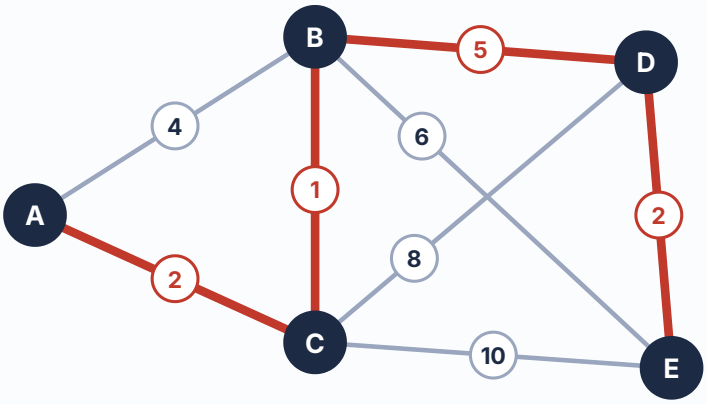
WORKED EXAMPLE

How many walks of length 2 go from A to D, and how many of length 3 go from A to B?

1. compute A^2 and A^3 on the GDC M1
2. $(A^2)_{AD}$ = **2** (A–B–D and A–C–D) A1
3. **$(A^3)_{AB}$ = 7** A1

- **Directed graphs** (and the **transition matrix** of a random walk, where each column shows the probabilities of moving from a vertex) are used for web page rankings: the long-term probabilities show which pages are visited most.

3 Minimum spanning trees HL



Kruskal: BC 1, AC 2, DE 2, BD 5 (skip AB: cycle)

Minimum spanning tree (red): weight 1 + 2 + 2 + 5 = 10.

Kruskal: sort the edges by weight and accept each one unless it makes a cycle.

WORKED EXAMPLE

Use Kruskal's algorithm to find a minimum spanning tree for the graph.

1. order: BC 1, AC 2, DE 2, AB 4, BD 5, BE 6, CD 8, CE 10 M1
2. accept BC, AC, DE; reject AB (it would make the cycle A–B–C) A1 R1
3. **accept BD: 4 edges for 5 vertices, done. Weight 10** A1

- **Prim's algorithm** (from a table): start at any vertex; repeatedly add the cheapest edge joining the tree to a new vertex. In matrix form, cross out the row of each new vertex and circle the smallest number in the columns of the tree.

COMMON MISTAKE

Stopping too early or too late. A spanning tree of n vertices has exactly **n – 1** edges.

4 Chinese postman problem HL

- **Method:** list the odd vertices; pair them up in every possible way; find the cheapest total of the shortest paths between pairs; repeat those edges. Route length = total weight + extra.

WORKED EXAMPLE

A street sweeper must travel every road in the graph and return to its start. Find the length of the shortest route.

1. total weight = 38 A1
2. odd vertices: D and E; shortest path DE = 2 M1
3. **38 + 2 = 40 (repeat DE)** A1

EXAM TIP

With four odd vertices there are three pairings, for example (A, B)(C, D), (A, C)(B, D), (A, D)(B, C). Find all three totals and choose the smallest.

- **Starting and finishing at different vertices:** keep the two odd vertices you would pair most expensively as the start and end, and pair the rest.

5 Travelling salesman problem HL

- **Table of least distances first:** if an edge is missing or long, replace it by the shortest path. For example, the least distance from A to B is 3 (A–C–B), not the direct 4.
- **Upper bound (nearest neighbour):** start at a vertex, always go to the nearest unvisited vertex, then return.
- **Lower bound (deleted vertex):** delete one vertex; find a minimum spanning tree of the rest; add the two shortest edges from the deleted vertex.

WORKED EXAMPLE

Find an upper bound from A by nearest neighbour, and a lower bound by deleting A.

1. least distances from A: C 2, B 3, D 8, E 9
2. nearest neighbour: A–C (2)–B (1)–D (5)–E (2)–A (9) = **19** M1 A1
3. delete A: MST of B, C, D, E = BC 1 + DE 2 + BD 5 = 8; add the two smallest least distances from A, 2 and 3: **8 + 5 = 13** M1 A1
4. **so 13 ≤ shortest tour ≤ 19** R1

COMMON MISTAKE

Using the nearest neighbour answer as "the" answer. It is only an **upper bound**: a tour of that length exists, but there might be a shorter one.

LINKING QUESTION

Adjacency and transition matrices use Matrices and eigenvalues; steady states of random walks link to Markov chains (Estimation, Poisson, hypothesis tests and Markov chains). Voronoi diagrams (Trigonometry, 3D geometry and Voronoi diagrams) are a geometric cousin of these network problems.

IDEA FOR YOUR IA / EXPLORATION

Model a real network (your school's corridors, a metro map, a delivery round) as a weighted graph and solve a Chinese postman or travelling salesman problem, then compare with the route people actually use.

Test yourself — cover the answers first

QUICK CHECK

1. HL How many edges does K_6 have?
2. HL How many edges does a tree on 9 vertices have?
3. HL A graph has degrees 2, 2, 4, 4. Is there an Eulerian circuit?
4. HL What does entry (i, j) of A^3 count?
5. HL Is nearest neighbour an upper or a lower bound for the TSP?

EXAM-STYLE

6. HL A graph has odd vertices P, Q, R, S with shortest paths PQ 5, RS 7, PR 6, QS 4, PS 9, QR 3, and total weight 80. Find the length of the shortest Chinese postman route.
7. HL Explain why the deleted vertex method gives a lower bound.

Show answers

1. 15
2. 8
3. yes, all degrees are even
4. walks of length 3 from vertex i to vertex j
5. an upper bound
6. (PQ, RS) = 12; (PR, QS) = 10; (PS, QR) = 12
   cheapest pairing PR + QS = 10
   80 + 10 = 90
7. any tour, with one vertex removed, leaves a path through the other vertices, which is a spanning tree, so it weighs at least the MST
   the tour also uses two edges at the deleted vertex, at least the two cheapest
   so every tour is at least MST + two cheapest edges

Stuck on a question?

Send it to me on WhatsApp. I read every message.

Share with a friend

Ask Riyaz Sir

Want regular help? Send this to your parent →