# DA - Graphs

|  |  |
| --- | --- |
| **Define edge/arc** | A line connecting two vertices. |
| **Define network/weighted graph** | A graph with weighted edges (i.e., edges with numbers). |
| **Define order/degree of vertex** | The number of edges starting and finishing at the vertex. |
| **Define loop** | An edge starting and finishing at the same vertex. |
| **Define weight** | A real world value assigned to an edge (e.g., metres). |
| **Define cycle** | A closed path. |
| **Define Hamiltonian cycle (with example)** | A route that...   * Visits every vertex exactly once * Returns to the starting vertex * Without repeating any edges   Example: ABCDEA for a pentagonal graph |
| **When is and isn’t a graph Hamiltonian-connected?** | * Is when there exists a Hamiltonian cycle for every vertex. * Isn’t when you’re forced through one vertex twice or thrice.   *If there exists a cycle for one then there must exist a cycle for them all (as you can merely reorder the letters).* |
| **No. of Hamiltonian Cycles** | * For distinct cycles in which you consider ABCA to equal ACBA, it is:      * So for non-distinct, it is:     *This is because if you had 4 vertices (in ABCDA), you would have 3! ways of arranging (A****BCD****A)****.*** *However, for any start it would be 4!.*  *Both ABCDA and ADCBA have the same shape but opposite directions. If you treat them as equal, you’ll get half as many cycles.* |
| **Define trial** | A route in which no edge is repeated (e.g., ABCEA). |
| **Define Eulerian trail** | A route which visits every edge EXACTLY ONCE. |
| **Define Eulerian cycle** | A trial which starts and ends at the same vertex **AND** visits every edge **EXACTLY ONCE**. |
| **TEPEV** | **T**rial **E**dge **P**ath **E**dge **V**ertex. |
| **Define connected graph** | Every node is connected to the graph.  *Therefore, it is possible to get from any node to any other node (however not necessarily directly).* |
| **Define simple graph** | A graph with no loops and no repeated edges. |
| **Define complete graph (Kn)** | A graph where every vertex is connected to every other vertex directly. |
| **What is the no. of edges on a complete graph?** | The number of edges on a complete graph is defined by:    *A complete graph has n - 1 outgoing edges and n edges in total thus you’d be tempted to say there are n(n - 1) edges. However, every edge is counted twice because every edge going out a vertex is going into another thus you divide by 2.* |
| **Define Eulerian graph** | A **CONNECTED** graph with **ONLY EVEN VERTICES**.  *This means you can start and end at the same point (because every vertex has an edge going into and out of it).* |
| **Define semi-Eulerian graph** | A **CONNECTED** graph with **EXACTLY TWO ODD VERTICES**.  *This means if you start at an odd degree vertex, you can visit every vertex yet you’ll end up at the other odd degree vertex.* |
| **Define non-Eulerian graph** | A **CONNECTED** graph with **MORE THAN TWO ODD VERTICES**. |
| **Define digraph** | A graph with one or more directed edges. |
| **Define bipartite graph** | Two sets of vertices with edges that can only connect from one set to the other. |
| **Define adjacency/incidence matrix** | A representation of the direct routes between the vertices of a graph in a matrix.  It is routes because…    Has 2 direct routes of getting from B to B.  Whereas…    Only has 1 direct route of getting from B to B. |
| **Define tree and a spanning tree** | * A tree is a simple graph with no cycles. * A spanning tree is a connected tree. |
| **How many edges are there in a spanning tree of n vertices and why?** | * n - 1 edges. * For each new vertex you add to the spanning tree, you increase the original tree (of 2 vertices and 1 edge) by 1 edge. |
| **Define minimum spanning tree** | A spanning tree of minimum weight. |
| **Define planar graph** | A graph that can be (re)drawn so that none of its arcs cross.  *This can be useful for insulated wires on microchips.* |
| **Define isomorphic graph** | A graph that has the same number of vertices connected in the same way as another. |
| **Define complement of a graph** | A graph of all missing edges required to make another graph complete. |
| **Define subgraph** | Part of a graph. |
| **Define subdivision** | Where you split an edge into two edges by adding an extra vertex. |
| **What is Euler's Formula and when does it hold true?** | Holds true for any **CONNECTED PLANAR** graph (e.g., many 3D solids).    *This can be remembered by using the vertices, faces, and edge of a cube.* |
| **What is Kuratowski’s Theorem?** | A graph is planar if and only if it does not contain a subgraph that is a subdivision of K5 or of K3,3. |
| **What is the triangle inequality?** |  |