CS Fundamentals
Matrices and graphs

Matrices

Matrix

- Matrix is a rectangular array of number, symbols or expressions.
- Arranged in rows and columns – two dimensional array.
- The individual items in a matrix are called its elements or entries.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

Matrix

- An m x n (read as “m by n”) matrix A is rectangular array of elements arranged into m rows and n columns.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{in} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mj} & \cdots & a_{mn} \end{bmatrix} \begin{matrix} \\ \\ \\ \leftarrow i\text{th row of A} \\ \\ \\ \end{matrix}$$

$$\begin{matrix} \uparrow \\ j\text{th column of A} \end{matrix}$$

Matrix

- Matrices are often referred to by their sizes.
- The size of a matrix is given in the form of a dimension – rows and columns.
- Since A has three rows and four columns, the size of A is 3 × 4

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \end{bmatrix}$$

Matrix Operations

- Addition and subtraction. Both matrices same dimensions.

$$A = \begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix} \qquad B = \begin{bmatrix} -1 & 2 \\ 0 & 3 \end{bmatrix}$$

$$A + B = \begin{bmatrix} 1 + (-1) & 3 + 2 \\ 2 + 0 & -1 + 3 \end{bmatrix} = \begin{bmatrix} 0 & 5 \\ 2 & 2 \end{bmatrix}$$

$$A - B = \begin{bmatrix} 1 - (-1) & 3 - 2 \\ 2 - 0 & -1 - 3 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 2 & -4 \end{bmatrix}$$

Matrix Operations

- Let’s practice!

$$\begin{bmatrix} 34 & 55 & 37 \\ 5 & 7 & 4 \\ 1 & 91 & 3 \end{bmatrix} - \begin{bmatrix} 1 & 4 & 122 \\ 2 & 51 & 7 \\ 3 & 6 & 8 \end{bmatrix}$$

Matrix Operations

- Scalar multiplication.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$2A = 2 \cdot \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 & 2 \cdot 2 & 2 \cdot 3 \\ 2 \cdot 4 & 2 \cdot 5 & 2 \cdot 6 \end{bmatrix} = \begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \end{bmatrix}$$

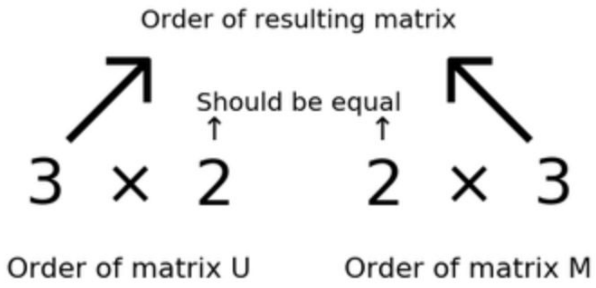
Matrix Operations

- Let’s practice! How much is 3xA

$$A = \begin{bmatrix} 5 & -2 & 8 \\ -4 & 1 & 6 \end{bmatrix}$$

Matrix Operations

- Dot product or matrix multiplication.



$$A = \begin{bmatrix} 1 & 3 \\ 2 & -1 \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 1 & 2 \\ 2 & 1 & -3 \end{bmatrix}$$

$$AB = \begin{bmatrix} (1 \cdot 0) + (3 \cdot 2) & (1 \cdot 1 + 3 \cdot 1) & (1 \cdot 2) + (3 \cdot -3) \\ (2 \cdot 0) + (-1 \cdot 2) & (2 \cdot 1) + (-1 \cdot 1) & (2 \cdot 2) + (-1 \cdot -3) \end{bmatrix}$$

$$AB = \begin{bmatrix} 6 & 4 & -7 \\ -2 & 1 & 7 \end{bmatrix}$$

Matrix Operations

- Let's practice!

$$\begin{bmatrix} 1 & 6 \\ 0 & 4 \\ 4 & 6 \end{bmatrix} \times \begin{bmatrix} 3 & 13 & -2 \\ 1 & -4 & 5 \end{bmatrix}$$

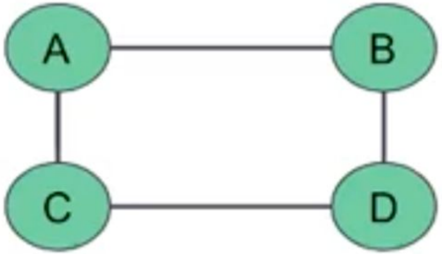
Graphs

Graphs

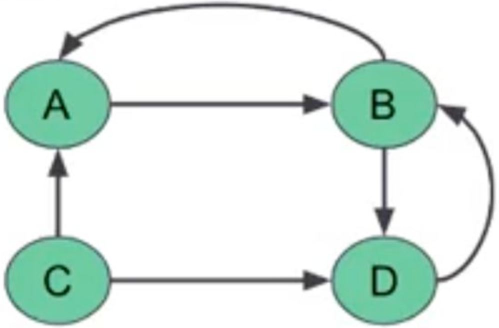
A **graph** is a representation of pairwise relationships between objects. The objects are represented as **nodes** and the relationships are represented by **edges**.

Directed vs Undirected Graphs

In an undirected graph, all relationships between objects are bi-directional.

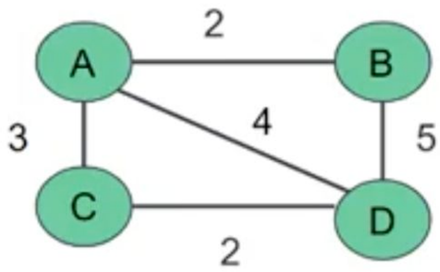


In a directed graph, relationships are explicitly noted with arrows.



Weighted Graphs

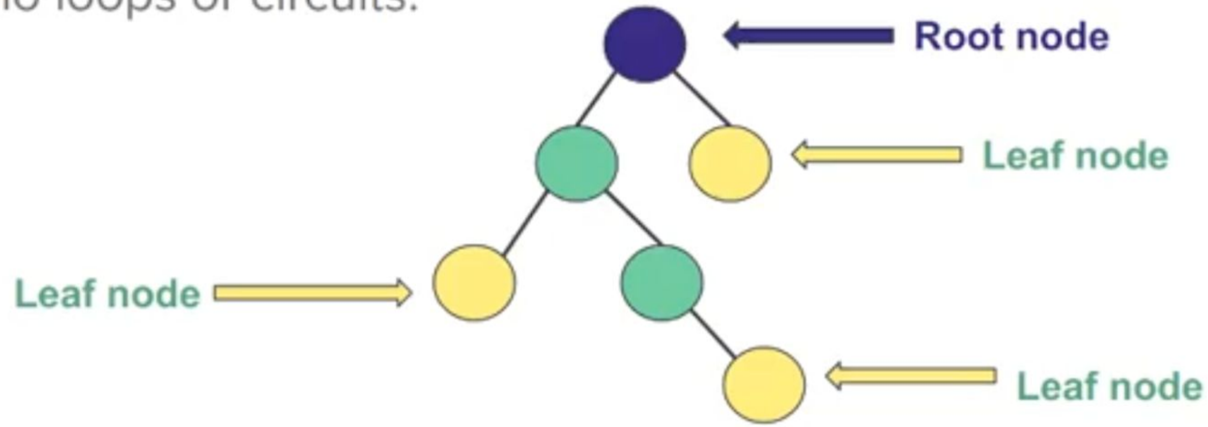
A weighted graph assigns values to edges. These values can be considered when finding the optimal path.



Tree

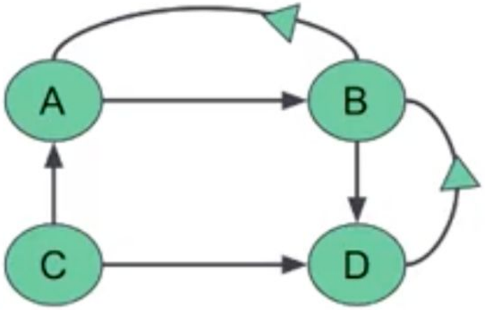
A tree is an undirected graph in which two vertices are connected by only one path, and every child node has only one parent. A **root node** has no parent. **Leaf nodes** have no children.

There are no loops or circuits.



Representing graphs, adjacency matrix

Make a table or matrix, with a row and column for each node. Indicate with 0 or 1 if an edge exists between each pair of nodes (where row is the origin and column is the destination of a directed edge).

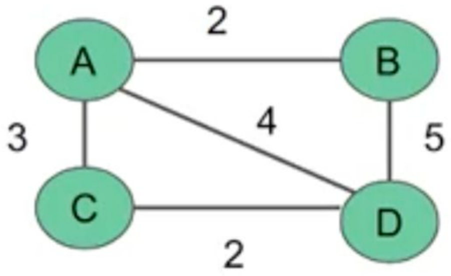


DESTINATION

ORIGIN	A	B	C	D
A	0	1	0	0
B	1	0	0	1
C	1	0	0	1
D	0	1	0	0

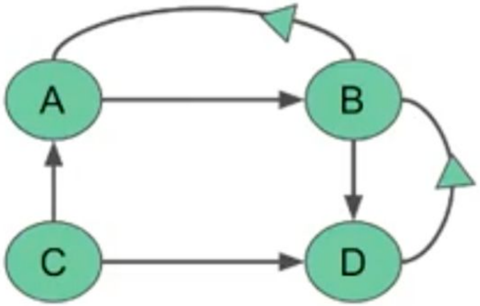
Weighted adjacency matrix

Instead of 0 and 1, use weights to specify connections.



	A	B	C	D
A	0	2	3	4
B	2	0	0	5
C	3	0	0	2
D	4	5	2	0

Edge lists

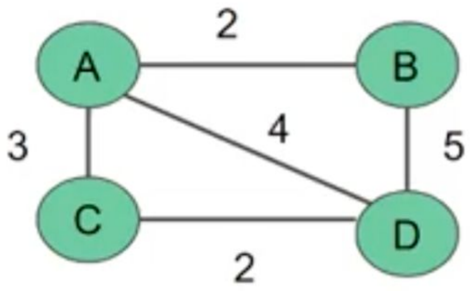


An **adjacency list** or **edge list** can also be used to represent a graph. In this example we can write an edge list as:

G = [AB, BA, BD, CA,CD, DB]

This is an example of a directed graph, or **digraph**. We could refer to the list as a **digraph edge list**.

Adjacency lists(weighted, undirected)



In an undirected graph, both directions are included for each edge, and weights are specified:

G = [A2B, B2A, A3C, C3A, A4D, D4A, B5D, D5B, C2D, D2C]

Note: You may also see the weights specified a bit differently, such as '2AB' or 'AB2'. It doesn't matter. What's important is that the left node symbol is the origin of the connection.