Graph Theory

Sunday, October 11, 2020 1:55 PM

Graphs.

Motivation. Many structures consist of

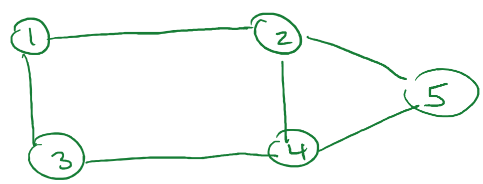
→ collection of objects

→ links between objects

Want framework for describing these objects and their properties.

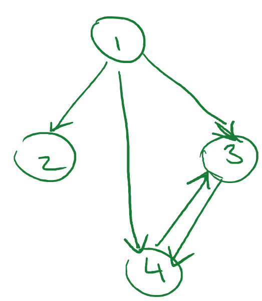
Defn. A graph consists of a set of <u>nodes/vertices</u> connected by <u>edges/arcs</u>.

ex.



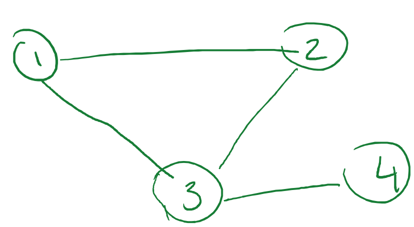
Classes. <u>Directed</u> - arrows represent uni-directionality

ex.



<u>Undirected</u> - no arrows, connection means mutual linkage

ex.

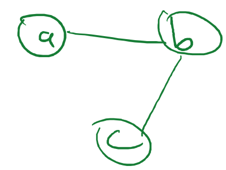


In CS103 - "graph" means undirected, finite graph

Formalizing graph declaration. A graph $G = (V, E)$ can be declared as (ex)

$V = \{a, b, c\}$

$E = \{\{a, b\}, \{b, c\}\}$



Note - for directed graphs, edges are represented w/ ordered pairs

Self-loops An edge from a node to itself is called a self-loop.
In undirected graphs, self-loops are generally not allowed.

Standard graph terminology.

Adjacency. Two nodes are adjacent if there is an edge linking them.

i.e., in a graph $G = (V, E)$, then two nodes $u, v \in V$ are adjacent if $\{u, v\} \in E$.

Path. A path in a graph $G = (V, E)$ is a sequence of one ~~or~~ more nodes $v_1, v_2, v_3, \ldots v_n$ where each v_i and v_{i+1} are adjacent.

The length of a path $v_1, v_2, \ldots, v_n$ is $n-1$ (i.e., the number of edges in the path).

Cycle A cycle in a graph is a path with length ≥ 2 from a node back to itself.

Simple path A simple path in a graph is a path that does not repeat any nodes or edges.

Simple cycle. A simple cycle is a cycle that does not repeat any nodes or edges except the first/last node.

Connectivity. Two nodes in a graph are connected if there exists a path between them.

A graph ~~G~~ is connected if all pairs of nodes in G are connected.

Connectivity is an equivalence relation.

Connected components.

Defn. Let $G = (V, E)$ be a graph. For each $v \in V$, the connected component containing v is the set

$$[v] = \{ x \in V \mid v \text{ is connected to } x \}$$