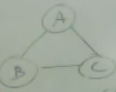
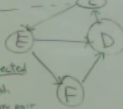
Graphs

A graph is represented by a collection of two sets. A set of vertices and a set of edges. There are two types of graphs, directed and undirected.

An undirected graph doesn’t have any restrictions on how you navigate the graph. For example the vertices would be {A, B, C} and the Edges would be {(A,B), (A,C), (B,C)}. The edges are unordered so they could also be put as {(B,A), (C,A), (C,B)}.



The other type, directed, has each edge associated with a direction. The difference is that now the edges are ordered pairs and cannot be rearanged like in an undirected. In this example you could navigate E--->D but you cannot go D----->E

The degree of a vertex is the number of lines intersecting to it. In the example above the indegree of D is 2 and the outdegree of D is 1.

A path is a sequence of verticies between vertex one and vertex two. For example EFD is a path connecting E to D. To calculate the length of the path you would name the first vertex (0) and the last one (n-1). In the example above the length of EFD is 3 because you have to traverse 3 verticies.

A cycle is a whole loop so instead of EFD it would be EFDCE. A cycle starts and ends on the same verticy.

A simple path is a path with no repeating verticies. A simple cycle is a cycle that only repeats the first value as the last (EFDCE).

A graph is considered connected if there is a path that connects every single verticy. A graph is considered complete if there is an edge connecting every single pair of verticies.

Adjacency Graph: Lists the origin and the path taken to connect the two verticies.

|  |  |
| --- | --- |
| C | E |
| D | C |
| E | D,F |
| F | D |