Graph: is a collection of objects with a relationship among the objects.

Leonard Euler

Tour the city by using each bridge once.

Graph theory

DEF: Graph = \( (V, E) \)

\( V \): set of objects = "Vertices"
\( E \): set of pairs \( \langle v, w \rangle \) where \( v, w \in V \)

"edges"

If \( E \) is reflexive: \( \langle v, w \rangle \in E \) iff \( \langle w, v \rangle \in E \)
The graph is "undirected"
Else the graph is called "directed" or "digraph"

Example:

\[ V = \{a, b, c, d\} \]
\[ E = \{\langle a, b \rangle, \langle b, c \rangle, \langle c, a \rangle, \langle b, d \rangle, \langle c, d \rangle\} \]

\( \langle c, D \rangle \)
Example:

\[ V = \{a, b, c, d\} \]
\[ E = \{(a, b), (b, c), (c, d), (d, a), (c, a), (a, c)\} \]

- The edge \( (v, v) \) is called a loop.
- A graph without loops is called simple.
- \( w \) is adjacent to \( v \) if \( (v, w) \in E \).
- The degree of a node \( v \) is the number of vertices adjacent to \( v \).
- The degree of a graph is the highest degree of a node in the graph.

- A path is a sequence of vertices \( \langle v_0, v_1, v_2, \ldots, v_n \rangle \) such that \( (v_i, v_{i+1}) \in E \).
  - \( n - 1 \) is the length of a path.
  - A path can be empty.
  - A path that does not contain a loop is called a "simple" path.

\[ a \rightarrow b \rightarrow c \rightarrow \alpha \]
\[ a \rightarrow b \rightarrow c \rightarrow \alpha \rightarrow e \]
\[ e \rightarrow d \rightarrow c \]
\[ a \rightarrow b \rightarrow c \rightarrow \alpha \]
\[ a \rightarrow b \rightarrow c \rightarrow \alpha \rightarrow e \]
\[ e \rightarrow d \rightarrow c \]
A cycle is a path \( \langle v_0, v_1, \ldots, v_n \rangle \) where \( v_0 = v_n \).

\[ (a, b, c, a) = \text{cycle}. \]
\[ (d, b, e, d) = \text{cycle}. \]
not a dag

\[ E \{ (a, b, \bar{a}) \} \]
not a dag

"Dag": Directed Acyclic Graph.

A graph is strongly connected if for every two vertices \( v, w \) different, there is a path \( \langle v, \ldots, w \rangle \).

Not strongly connected but weakly connected.

not strongly connected
not weakly connected.
Strongly Connected

- A graph is weakly connected if for every two different vertices \( u, v \), either there is a path \( \langle u, \ldots, v \rangle \) or a path \( \langle v, \ldots, u \rangle \).

\[ \equiv \text{OPERATIONS} \equiv \]
- \text{add Vertex } (G, v)
- \text{add Edge } (G, e) \quad e = \langle v, w \rangle
- \text{neighbors } (G, v) = \text{List of vertices adjacent to } v

\text{Data Structures for Graphs:}

"the Adjacency Matrix"

\[
\begin{array}{cccccc}
\text{from} & a & b & c & d & e \\
\hline
a & 0 & 1 & 0 & 1 & 0 \\
b & 0 & 0 & 0 & 1 & 0 \\
c & 0 & 1 & 0 & 0 & 1 \\
d & 0 & 0 & 1 & 0 & 1 \\
e & 0 & 0 & 1 & 0 & 0 \\
f & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\]

\text{add Edge = cheap}
\text{add Node = Medium}
\text{neighbors = Medium}

\text{Directed Graph}

\[
\begin{array}{ccccccc}
\text{from} & a & b & c & d & e \\
\hline
a & 0 & 1 & 0 & 1 & 0 \\
b & 0 & 0 & 0 & 1 & 0 \\
c & 0 & 1 & 0 & 0 & 1 \\
d & 0 & 0 & 1 & 0 & 1 \\
e & 0 & 0 & 1 & 0 & 0 \\
f & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\]

\text{Undirected Graph}

\[
\begin{array}{ccccccc}
\text{from} & a & b & c & d & e \\
\hline
a & 0 & 1 & 1 & 0 & 0 \\
b & 1 & 0 & 1 & 1 & 0 \\
c & 1 & 1 & 0 & 1 & 0 \\
d & 0 & 0 & 1 & 0 & 1 \\
e & 0 & 0 & 1 & 0 & 0 \\
e & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\]

\text{Weighted Graph}

\[
\begin{array}{cccc}
\text{from} & a & b & c \\
\hline
a & 0 & 5 & 8 \\
b & 3 & 0 & 3 \\
c & 8 & 3 & 0 \\
\end{array}
\]
Pro: Simple, 

Cons: Memory expensive. (for the zeroes) Neighbors is expensive

Example

- highest degree of a vertex 6
- A city graph can have 10,000 vertices.

Adjacency List Data Structure

Directed

Undirected

```
add Node cheap
add Vertex cheap
Neighbors cheap.
```

"Is a a neighbor of b?" = expensive
Is a a neighbor of u?

weighted

Data Structures Page 6