Graph Basics

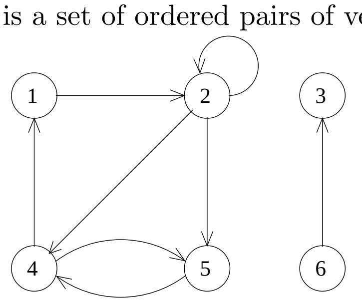
Module 5: Graphs

Overview

Once you learn about graphs, you realize that graphs are everywhere around us. We can model pretty much anything with a graph! In this lecture we introduce the concept of *graph*, talk about graph representation and introduce basic basic problems on graphs.

What is a graph?

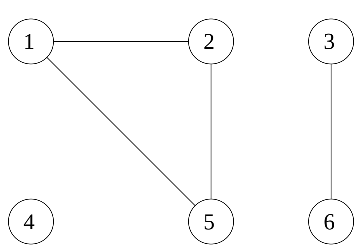
- A graph $G = (V, E)$ consists of a set of *vertices* V and a set of *edges* E
- Vertices are objects and edges model relationships
- E.g: graphs can model a network of people, where edges mean friendships
- There are two types of graphs: directed and undirected
- *Directed graph*: E is a set of ordered pairs of vertices (u, v) where $u, v \in V$



V = {1, 2, 3, 4, 5, 6}

E = {(1,2), (2,2), (2,4), (2,5)
(4,1), (4,5), (5,4), (6,3)

- *Undirected graph*: E is a set of unordered pairs of vertices $\{u, v\}$ where $u, v \in V$



V = {1, 2, 3, 4, 5, 6}

E = {{1,2}, {1,5}, {2,5}, {3,6}}

- Note: Some definitions do not allow self loops (because graphs are simpler to work with).
- Often the edges in a graph have weights (e.g. road networks have distances); more on this later

Examples

- Graphs come up everywhere!!
- E.g.: road networks (vertices are intersections, edges are streets); social graphs (vertices are people, edges are “friends” relationshipis); WWW graph (webpages and hyperlinks); twitter graph (people/accounts and the ”follow” relations); internet (servers and cables); financial graphs (stocks and transactions); neural networks (neurons, synapses); and many, many more
- Some example of questions we can ask on graphs:
  - Is it directed or undirected? (if I am your friend, does it mean you are my friend?)
  - Are two people friends? How close?
  - Who is a “star”? What is a “star”?
  - Am I linked by some chain of friends to a “star”?
  - Is there a path between any two people in the graph?
  - Who has the most friends?
  - What is the largest clique (i.e. group of friends such that everyone is friends with everyone else)?
  - What is the shortest route between two vertices?
  - Can a web crawler reach the whole WWW?
  - If a certain server goes down, does that disconnect the network?

More Definitions and Terminology

Undirected graphs

- Edge (u, v) is said to be *incident* on u and v
- The *degree* of vertex is the number of edges incident to it.
- Paths: A *path* from u_1 to u_2 is a sequence of vertices $< u_1{=}v_0, v_1, v_2, \cdots, v_k{=}u_2 >$ such that $(v_i, v_{i+1}) \in E$. If $v_0 = v_k$ then it is a *cycle*. The length of a path is the number of edges on it.
- Two vertices are *connected* if there is a path between them in G
- Connectivity (in undirected graphs only): An undirected graph is *connected* if every pair of vertices are connected by a path
- A graph consists of connected components: A *connected component* of an undirected graph is a set of vertices such that any two vertices are connected by a path.
- An undirected graph is called a *tree* if there is a single path between any two vertices in G.
- It can be shown that a tree is connected, has no cycles, and the number of edges is precisely $|E| = |V| - 1$ (actually it can be shown that any two of these properties implies the third one).

Directed graphs (digraphs)

- A directed edge (u, v) is outgoing from u and incoming into v
- *In (out) degree* of a vertex is the number of edges entering (leaving) it.
- Paths: A *path* from u_1 to u_2 is a sequence of vertices $< u_1{=}v_0, v_1, v_2, \cdots, v_k{=}u_2 >$ such that $\{v_i, v_{i+1}\} \in E$. If $v_0 = v_k$ then it is a *cycle*. The length of a path is the number of edges on it.
- Reachability: if there is a path from u_1 to u_2 we write $u_1 \rightsquigarrow u_2$ and we say that u_2 is *reachable* from u_1
- A directed graph is *strongly connected* if every pair of vertices are reachable from each other
- A digraph consists of strongly connected components: A *strongly connected component* of a graph is a set of vertices such that between any two vertices are mutually reachable from each other

Graph Size

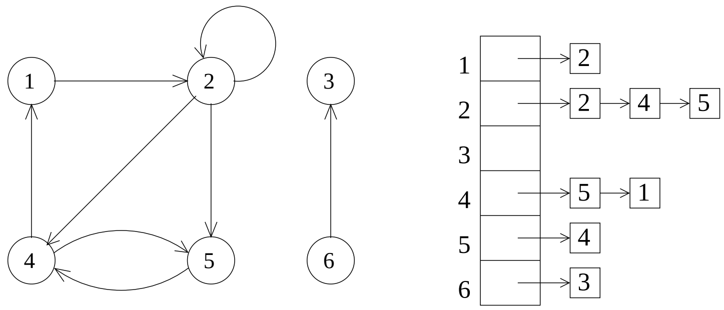
- We will express the space and time complexity of graph algorithms in terms of $|V|$ and $|E|$, often dropping the $|$'s.
- The standard notation is $|V| = n$ and $|E| = m$
- From an analysis point of view, it is important to understand the relation between $|V|$ and $|E|$, and their bounds.
- The number of edges in a graph can be as little as 0: $|E| \geq 0$
- The largest number of edges in a graph is $|E| = O(|V|^2)$. The exact count depends on whether the graph is directed or not, and if self loops are allowed.
- Remember that a tree of V vertices has $|E| = |V| - 1$ edges; that's $|E| = \Theta(|V|)$
- In general, if $|E| = \Theta(|V|^2)$ the graph is said to be dense
- If $|E| = \Theta(|V|)$ the graph is said to be sparse

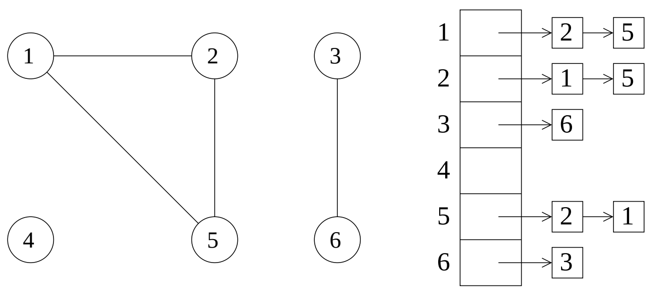
Graph representation

A graph can be represented as an adjacency list or adjacency matrix.

- *Adjacency-list* representation: Array of size $|V|$, each vertex stores its list of outgoing edges

  Examples:



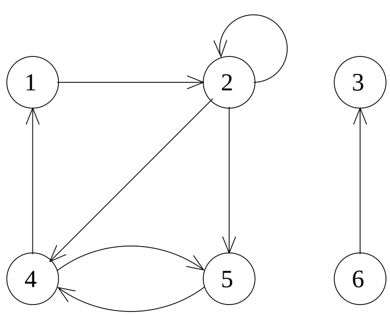


  - Note: For undirected graphs, every edge is stored twice.
  - If graph is weighted, a weight is stored with each edge.
  - Space: $\Theta(|V| + |E|)$

- *Adjacency-matrix* representation: $|V| \times |V|$ matrix A where

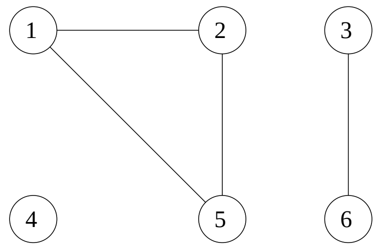
$$a_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

  Examples:



J

i	1	2	3	4	5	6
1	0	1	0	0	0	0
2	0	1	0	1	1	0
3	0	0	0	0	0	0
4	1	0	0	0	1	0
5	0	0	0	1	0	0
6	0	0	1	0	0	0



j

i	1	2	3	4	5	6
1	0	1	0	0	1	0
2	1	0	0	0	1	0
3	0	0	0	0	0	1
4	0	0	0	0	0	0
5	1	1	0	0	0	0
6	0	0	1	0	0	0

  - Note: For undirected graphs, the adjacency matrix is symmetric along the main diagonal ($A^T = A$).
  - If graph is weighted, weights are stored instead of one's.
  - Space: $\Theta(|V|^2)$

- Comparison of matrix and list representation:

Adjacency list	Adjacency matrix
$\Theta(\|V\| + \|E\|)$ space	$\Theta(\|V\|^2)$ space
Good if graph *sparse* ($\|E\| << \|V\|^2$)	Good if graph *dense* ($\|E\| \approx \|V\|^2$)
No quick access to (u, v)	$O(1)$ access to (u, v)

- We will use adjacency list representation unless stated otherwise ($\Theta(|V| + |E|)$ space).

- Interestingly, many/most large graphs in real-life are sparse (internet, social networks, etc).