Graphs

- Graphs are everywhere
- Types and Terminology: Handshaking lemma
- Representation, Complement, Transpose, Subgraph
- Walks, Paths and Cycles
- (Strongly) Connected and k-Connected graphs
- Applications: BFS, DFS, Eulerian graphs
- Advanced Applications: Optimization & Massive Graph Analysis

Imdad ullah Khan

Graph Representation: Adjacency Matrix

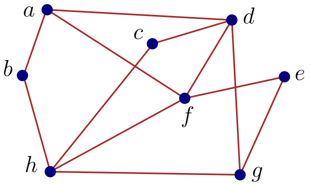
Undirected Simple Graphs

$G = (V, E)$

V is set of vertices

E is set of edges

(unordered pairs (2-subsets) of V)



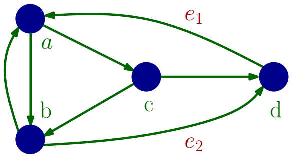
Directed Graphs (digraphs)

$G = (V, E)$

V is set of vertices

E is set of edges

(ordered pairs of V)

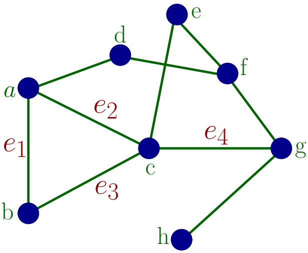


Undirected Graph Representation: Adjacency Matrix

Represent undirected $G = (V, E)$ with an **adjacency matrix** A_G

- Fix an arbitrary ordering of V
- One row for each vertex in V
- One column for each vertex in V

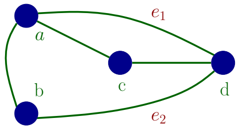
$$A_{ij} = \begin{cases} 1 & if (v_i, v_j) \in E \\ 0 & if (v_i, v_j) \notin E \end{cases}$$



$A_G =$

	a	b	c	d	e	f	g	h
a	0	1	1	1	0	0	0	0
b	1	0	1	0	0	0	0	0
c	1	1	0	0	1	0	1	0
d	1	0	0	0	0	1	0	0
e	0	0	1	0	0	1	0	0
f	0	0	0	1	1	0	1	0
g	0	0	1	0	0	1	0	1
h	0	0	0	0	0	0	1	0

Undirected Graph Representation: Adjacency Matrix



$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Let A be the adjacency matrix of a simple graph graph on n vertices.

ICP 14-22 How many entries are there in A?

ICP 14-23 How many 1's are there in A?

ICP 14-24 How many 0's are there in A?

ICP 14-25 What are diagonal entries of A?

ICP 14-26 How many 1's are there in row corresponding to vertex v?

ICP 14-27 How many 1's are there in column corresponding to v?

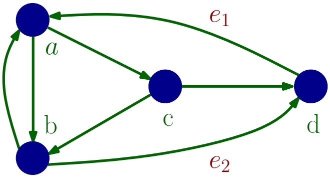
Directed Graph Representation: Adjacency Matrix

Digraph $G = (V, E)$ is a relation on V

Represent G with an **adjacency matrix** A_G

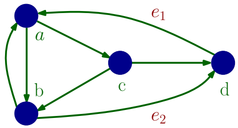
- Fix an arbitrary ordering of V
- One row for each vertex in V
- One column for each vertex in V

$$A_{ij} = \begin{cases} 1 & if (v_i, v_j) \in E \\ 0 & if (v_i, v_j) \notin E \end{cases}$$



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Directed Graph Representation: Adjacency Matrix



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Let A be the adjacency matrix of a simple digraph graph on n vertices.

ICP 14-28 How many entries are there in A?

ICP 14-29 How many 1's are there in A?

ICP 14-30 How many 0's are there in A?

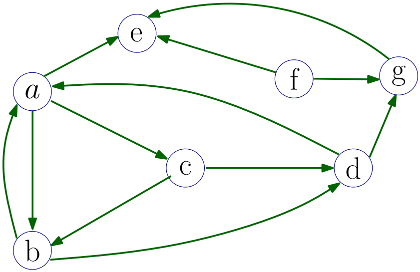
ICP 14-31 What are diagonal entries of A?

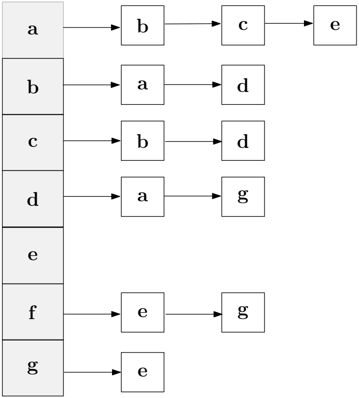
ICP 14-32 How many 1's are there in row corresponding to vertex v?

ICP 14-33 How many 1's are there in column corresponding to v?

Graph Representation: Adjacency List

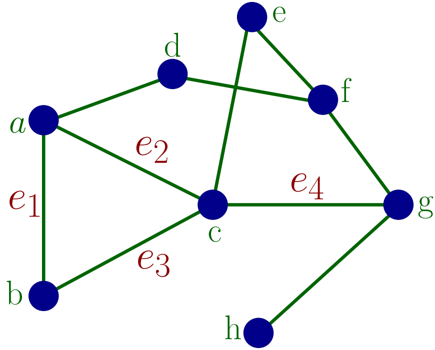
Represent digraph by listing neighbors of each vertex

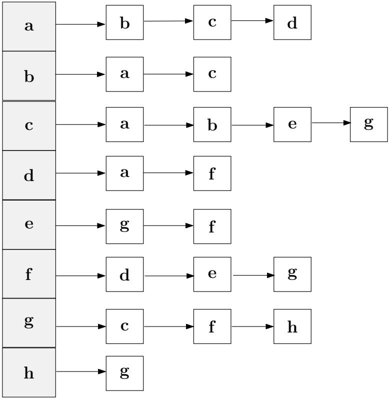




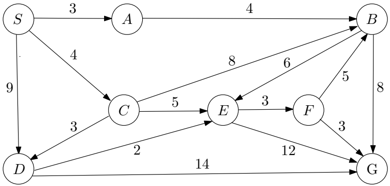
Graph Representation: Adjacency List

Represent undirected graph by listing neighbors of each vertex





Weighted Graph Representation



Weighted Adjacency Matrix

	S	A	B	C	D	E	F	G
S	0	3	0	4	9	0	0	0
A	0	0	4	0	0	0	0	0
B	0	0	0	0	0	6	0	8
C	⋮							
D								
E								
F								
G								

Weighted Adjacency Lists

- **S** → **A** 3 → **C** 4 → **D** 9
- **A** → **B** 4
- **B** → **E** 6 → **G** 8
- **C** → **B** 8 → **D** 3 → **E** 5
- **D** → **E** 2 → **G** 14
- **E** → **F** 3 → **G** 12
- **F** → **B** 5 → **G** 3
- **G**

Graph Representation: Tradeoff

$G = (V, E), \quad |V| = n, \quad |E| = m$

- Adjacency matrix representation
  - requires n^2 bits
  - Edge query $\big[(a, b) \in E\,?\big]$ requires one memory lookup
- Adjacency list representation
  - requires $2m$ integers (vertex ids) $\sim 2m \log n$ bits
  - Edge query $\big[(a, b) \in E\,?\big]$ requires list traversal

Usually real-world graphs are very **sparse** $m \ = \ C \cdot n \log n$

▷ So adjacency lists are preferred

For very **dense** graphs adjacency matrix is better

Graph Complement

Graph Complement

$G = (V, E) \rightarrow \overline{G} = (V, \overline{E})$

$$(u, v) \in \overline{E} \;\; iff \;\; (u, v) \notin E$$

- Vertex set is the same
- Each edge become non-edge and each non-edge becomes edge

▷ (except self-loops)

ICP 14-34 How to compute $\overline{G}$ from adjacency matrix and list of G ?

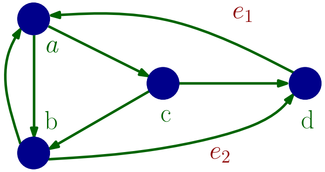
Graph Transpose

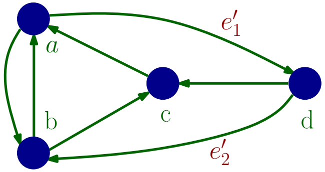
Graph Transpose

$G = (V, E) \rightarrow G^T = (V, E')$ ▷ G is a diagraph

$$(u, v) \in E' \;\; iff \;\; (v, u) \in E$$

- Vertex set is the same
- Direction/orientation of edges are reversed





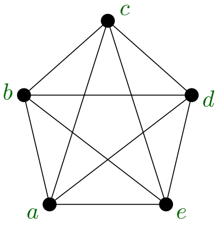
ICP 14-35 How to compute G^T from adjacency matrix and list of G ?

Subgraph

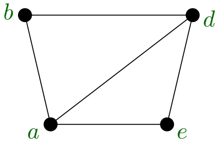
$H = \underline{(V', E')}$ is a **subgraph** of $G = \underline{(V, E)}$, if

$$V' \subseteq V \quad \text{AND} \quad E' \subseteq E$$

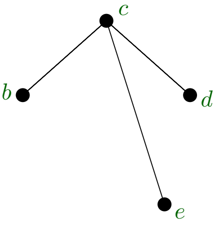
Denoted as $H \subseteq G$



G



$H_1 \subseteq G$

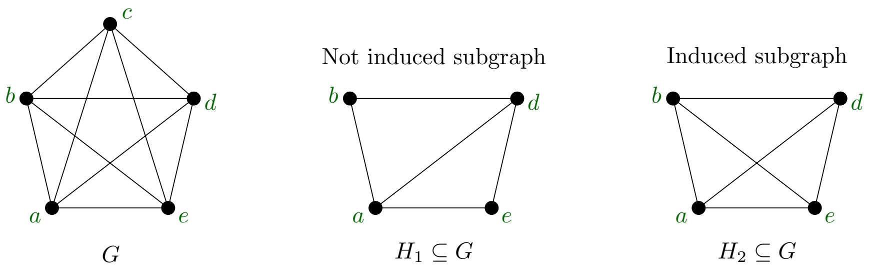


$H_2 \subseteq G$

Induced Subgraph

$H = \underline{(V', E')}$ is a **induced subgraph** of $G = \underline{(V, E)}$, if

$V' \subseteq V$ AND $E' = E|_{V'}$ (all edges in E with both endpoints in V')



An induced subgraph is completely determined by V'

Spanning Subgraph

$H = \underline{(V', E')}$ is a **spanning subgraph** of $G = \underline{(V, E)}$, if

$$V' = V \quad \text{AND} \quad E' \subseteq E$$

Denoted as $H \subseteq G$