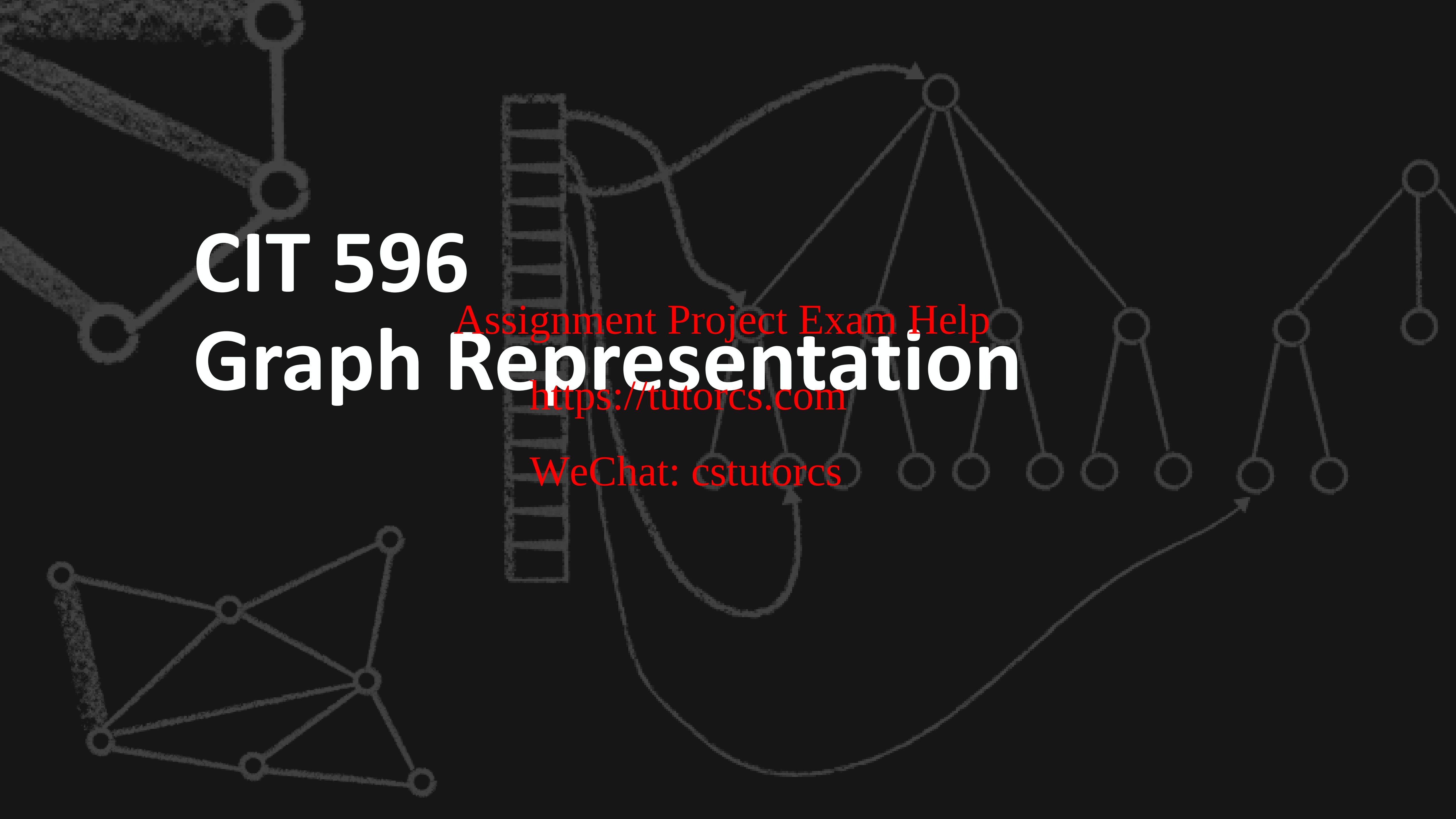
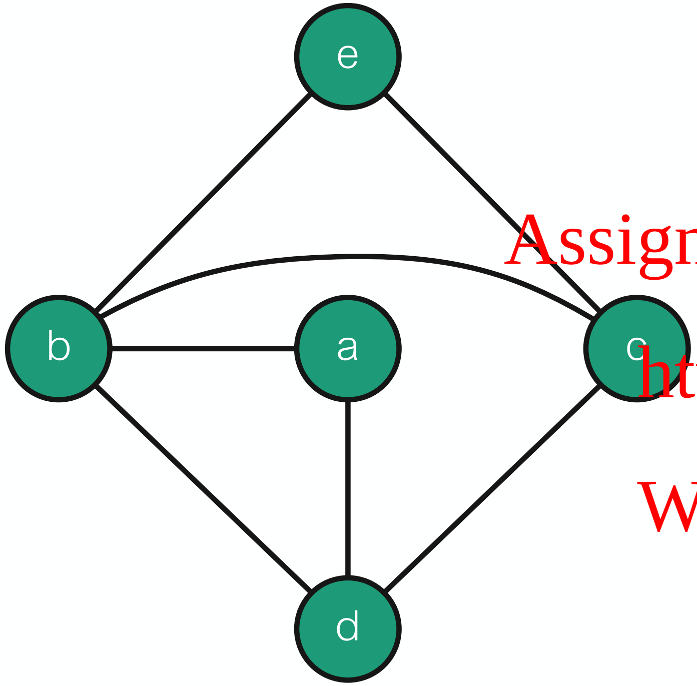


CIT 596
Graph Representation

HOW TO REPRESENT GRAPHS

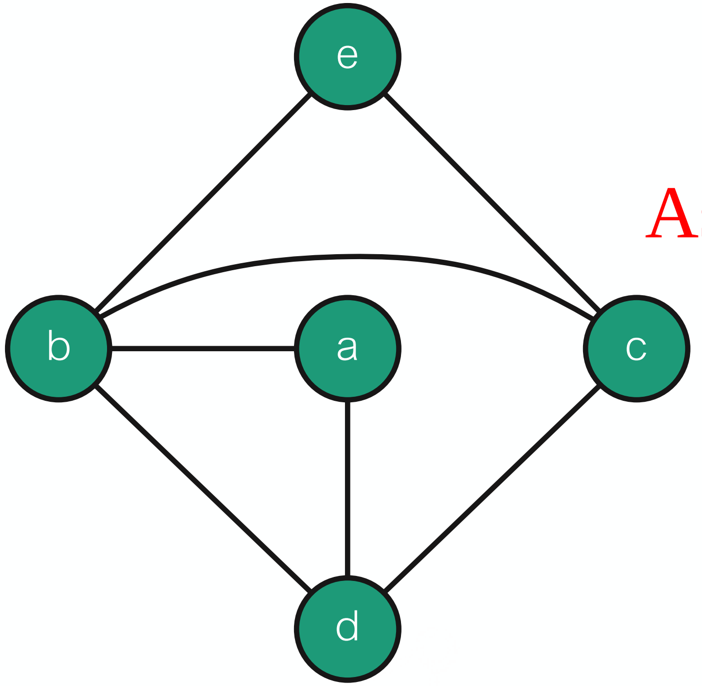


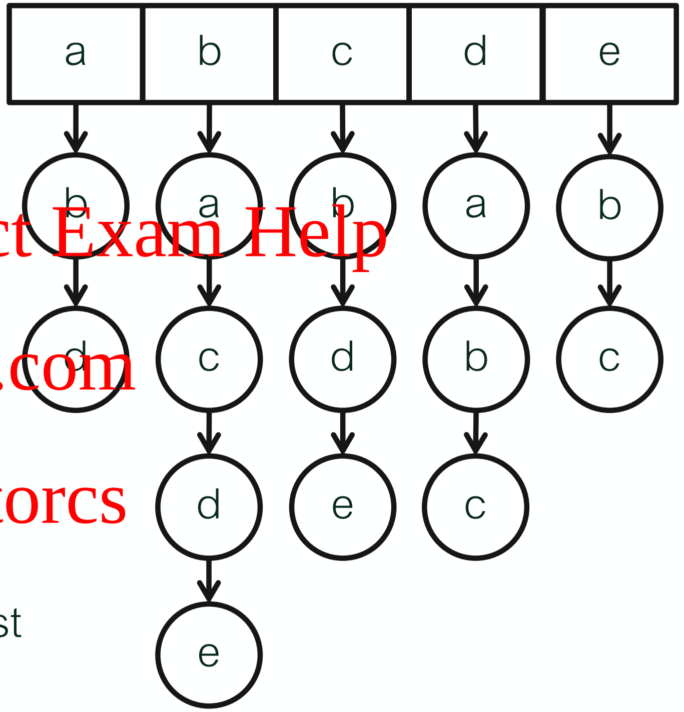
(Symmetric) Adjacency Matrix

	a	b	c	d	e
a	0	1	0	1	0
b	1	0	1	1	1
c	0	1	0	1	1
d	1	1	1	0	0
e	0	1	1	0	0

- Space required? $n \times n = O(n^2)$
- Checking if an edge (u, v) exists? $O(1)$
- Finding all neighbors of a node u? $O(n)$

HOW TO REPRESENT GRAPHS





Adjacency List

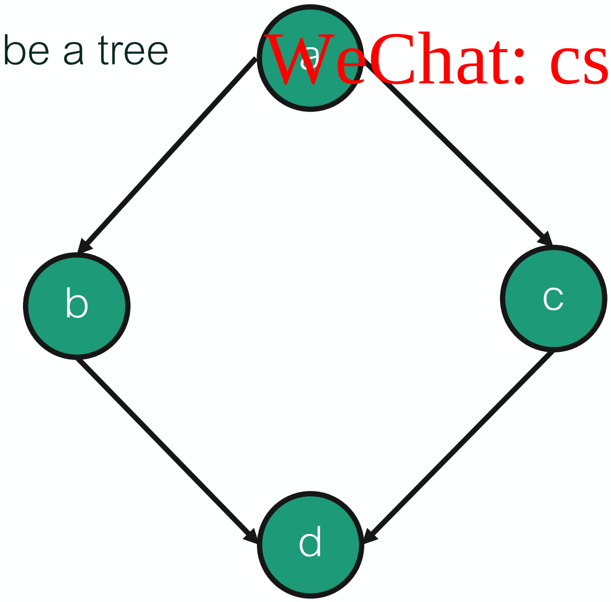
- Space required? $O(n+m)$
- Checking if an edge (u, v) exists? $O(\deg(u))$
- Finding all neighbors of a node u? $O(\deg(u))$

GRAPH INTERFACE

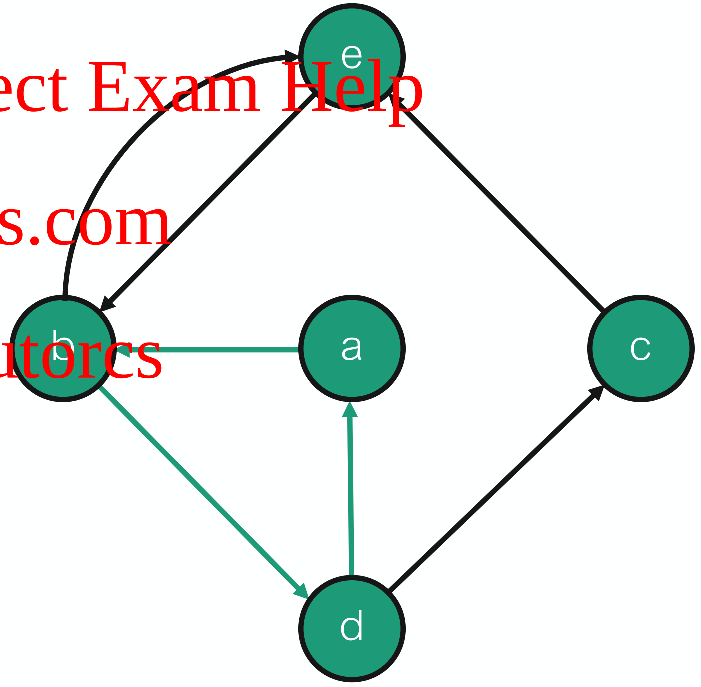
- Create a graph by specifying list of vertices and adding edges sequentially
  - $O(n+m)$ for adjacency list representation
  - $O(n^2)$ for adjacency matrix representation
- Visit all neighbors of a given node v
  - $O(\deg(v))$ for adjacency list representation
  - $O(n)$ for adjacency matrix representation
- Test if some pair (u, v) is adjacent (rarely used)
  - $O(\deg(v))$ for adjacency list representation
  - $O(1)$ for adjacency matrix representation
- Other operations
  - Sort edges by weight, find edge of minimum weight in a set
    - can be built up from primitive operations above

DIRECTED GRAPHS

- Edges are directed
  - Thought of as ordered pairs
  - $(a, b) \neq (b, a)$
- Directed Acyclic Graph (DAG)
  - A directed graph with no cycles
  - Does not have to be a tree



This graph contains cycles



DIRECTED GRAPH FACTS

- Paths and cycles must respect edge directions.
- It’s possible that vertex u can reach v but not vice versa.
- Define symmetric “can reach” relation R: $(u, v) \in R$ if u can reach v and v can reach u.
- R is an equivalence relation!
- Equivalence classes of R are called strongly connected components (SCC)
- In DAGs, every vertex is in an SCC by itself.
- The interface for directed graphs is similar to the interface for undirected graphs.