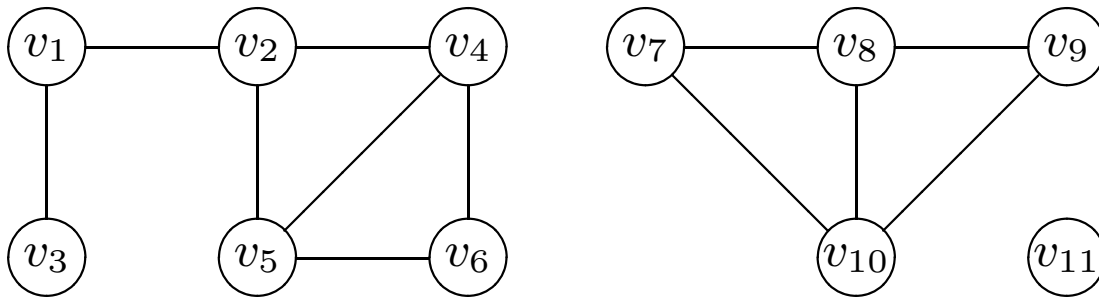


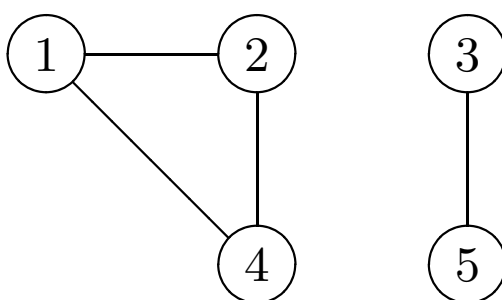
## Definitions



You should be familiar with:

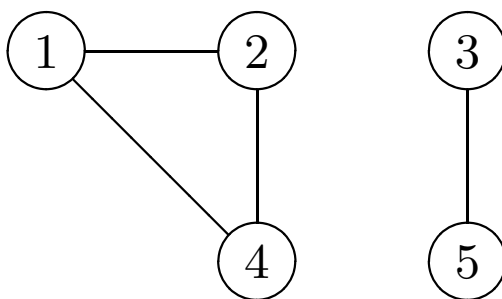
- Graph,  $V$ ,  $E$ ,  $n = |V|$ ,  $m = |E|$ .
- Vertices, adjacent, neighbours.
- Edges, incident, degree,  $\sum_{v \in V} \deg(v) = 2m$ .
- Multi-edge, loop, simple graph.
- Path, cycle, connected.
- Breadth-First search (we will review it again.)

# Adjacency Matrix

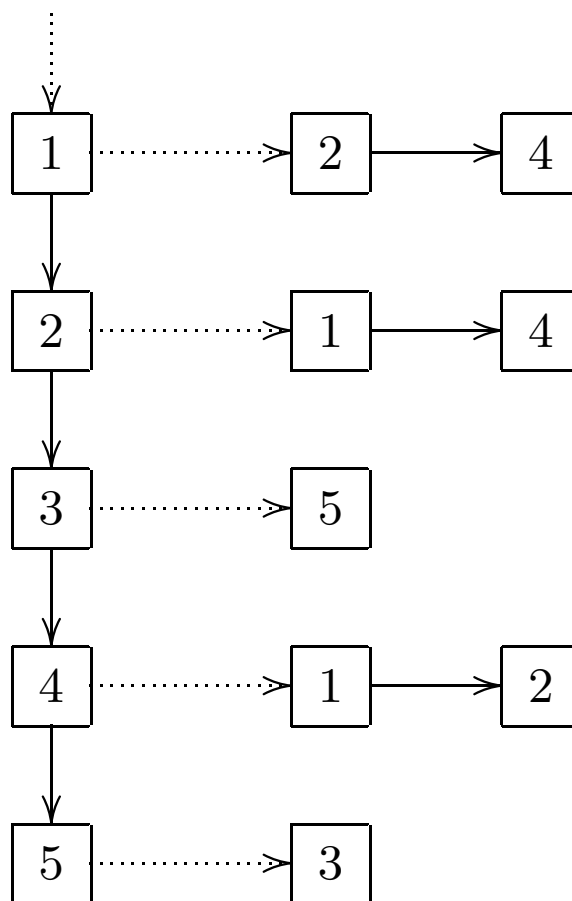


A	1	2	3	4	5
1	0	1	0	1	0
2	1	0	0	1	0
3	0	0	0	0	1
4	1	1	0	0	0
5	0	0	1	0	0

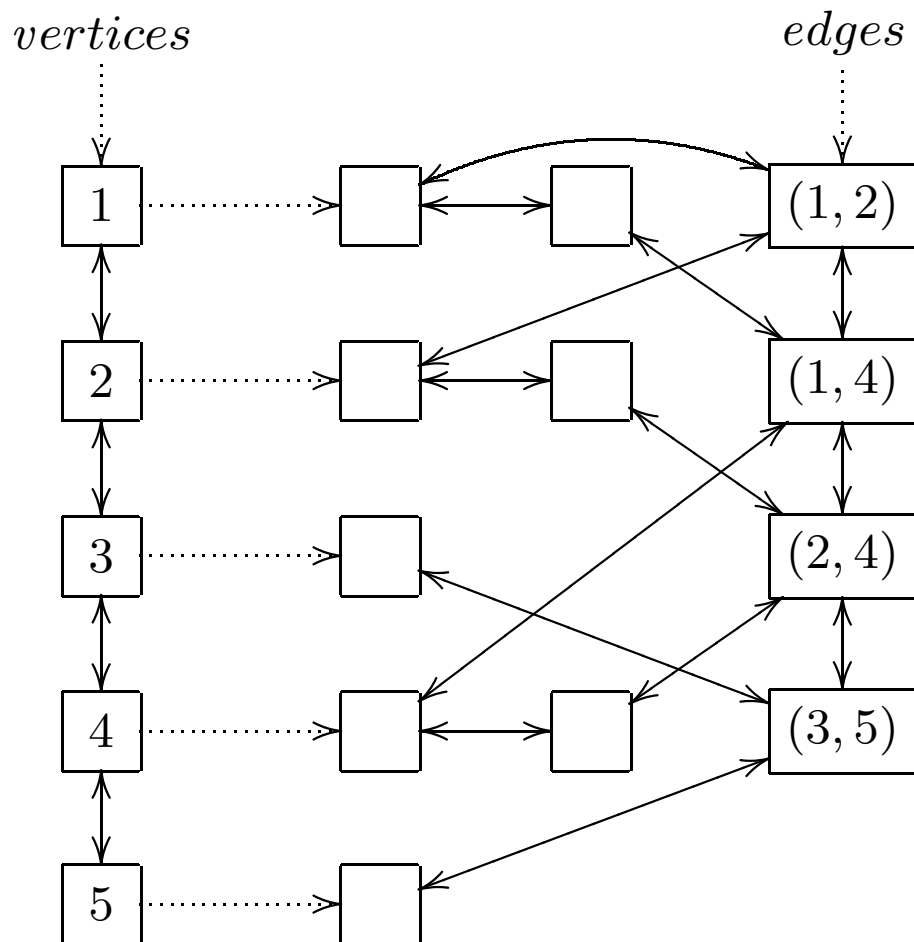
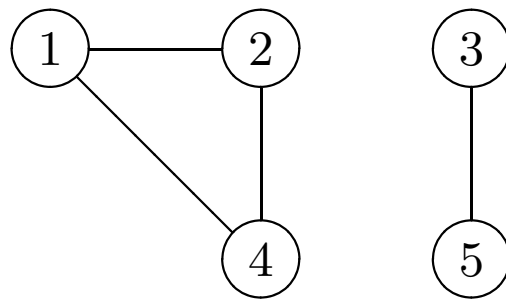
# Adjacency Lists



*vertices*



# Incidence Lists



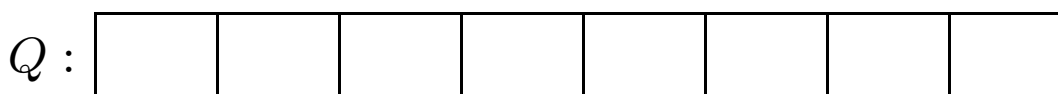
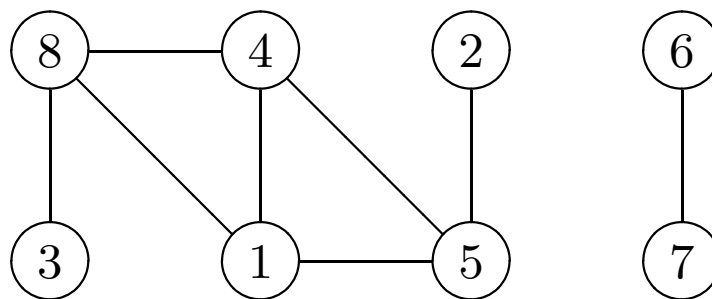
(plus references from edges to their endpoints.)

# Breadth-First Search

**Algorithm** BFS(Graph  $G$ , vertex  $s$ ):

- 1: Initialize an empty queue  $Q$ .
- 2: Color all vertices white.
- 3: Color  $s$  gray and put it into  $Q$ .
- 4: **while**  $Q$  is not empty **do**
- 5:      $u \leftarrow Q.dequeue()$
- 6:     **for** all neighbours  $v$  of  $u$  **do**
- 7:         **if**  $v$  is white **then**
- 8:             Color  $v$  gray and put it into  $Q$ .
- 9:     Colour  $u$  black.

## Example: BFS(8)



## Depth-First Search

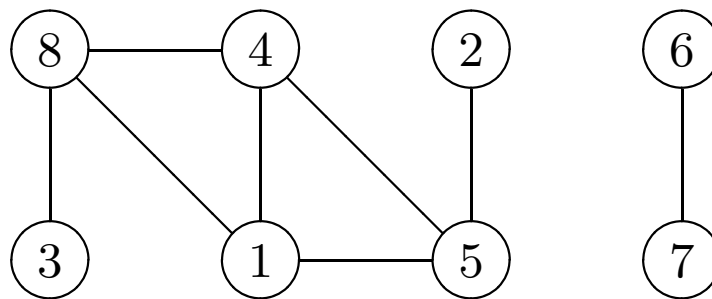
**Algorithm** DFS(Graph  $G$ ):

- 1: Colour all vertices white.
- 2: **for** all vertices  $v$  in  $G$  **do**
- 3:   If  $v$  is white, DFS-Visit( $v$ )

**Algorithm** DFS-VISIT(Vertex  $u$ ):

- 1: Colour  $u$  gray.
- 2: **for** all neighbours  $v$  of  $u$  **do**
- 3:   **if**  $v$  is white **then**
- 4:     DFS-Visit( $v$ )
- 5: Colour  $u$  black.

Example:



DFS-Visit called with

--	--	--	--	--	--	--	--

# Depth-First Search

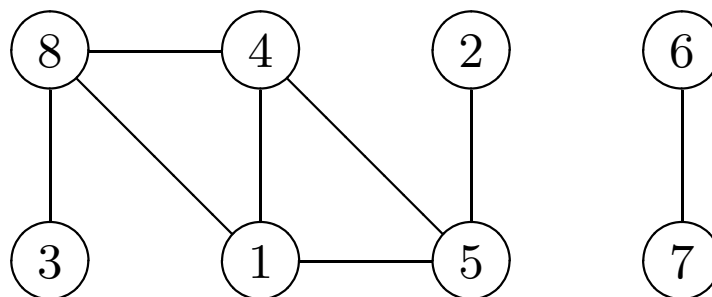
**Algorithm** DFS-NonRecursive(Graph  $G$ ):

- ```

1: Colour all vertices white and put them on a stack  $S$ .
2: while  $S$  is not empty do
3:    $u \leftarrow S.\text{peek}()$ 
4:   if  $u$  is white then
5:     Colour  $u$  gray.
6:     for all neighbours  $v$  of  $u$  do
7:       if  $v$  is white then
8:          $S.\text{push}(v)$ 
9:   else
10:    Colour  $u$  black.
11:     $S.\text{pop}()$ 

```

Example:



S:

[illegible]