Cycles and Girth

Instructor: Meng-Fen Chiang

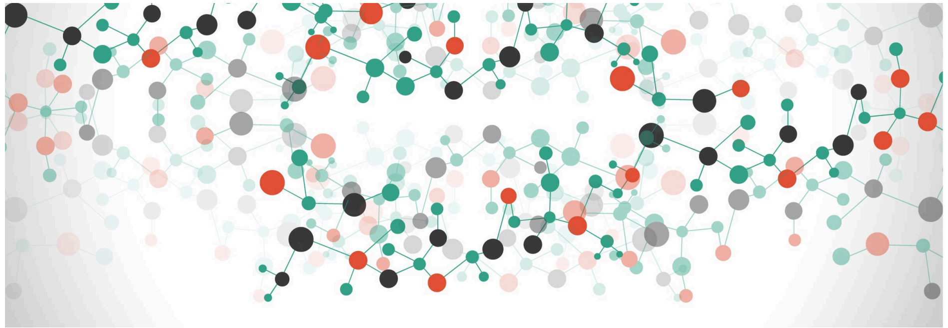
COMPCSI220: WEEK 10

THE UNIVERSITY OF AUCKLAND
Te Whare Wananga o Tamaki Makaurau
NEW ZEALAND

Slides adapted from Mark Wilson, Georgy Gimel'farb, Simone Linz and Tanya Gvozdeva

OUTLINE

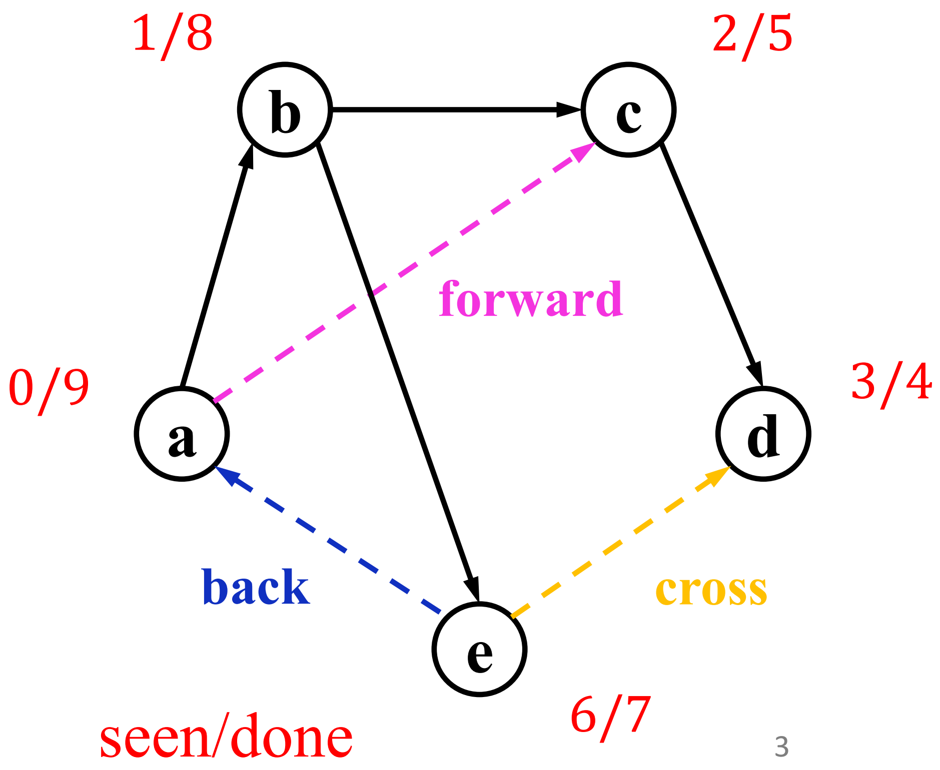
- Terminology
  - Cycle
  - Girth
- DFS Application: Cycle Detection
- BFS Application: Girth Detection
- Correctness of Girth Computation



RECAP: Example 23.7

- Explain how to determine (u,v) in DFS algorithm whether it is a tree-, back-, forward- or cross-arc?

1. If v is white, then (u,v) is a **tree** arc
2. If v is grey, then (u,v) is a **back** arc
3. If v is black, then (u,v) is
   - a **cross** arc($seen[v]<seen[u]$, $done[v]<seen[u]$), or
   - a **forward** arc($seen[u]<seen[v]<done[v]<done[u]$).



Cycle Detection

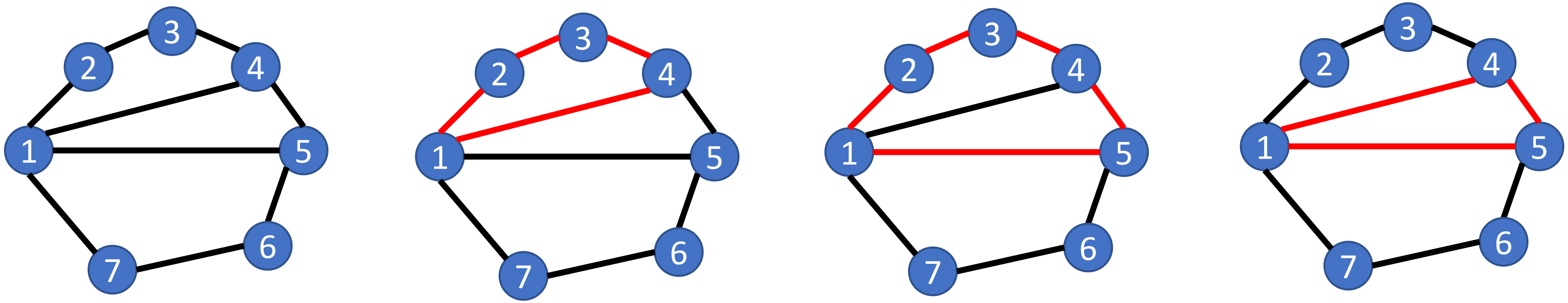
- Suppose that there is a cycle in G and let v be the node in the cycle visited first by DFS. If (u, v) is an arc in the cycle then it must be a back arc.

- Conversely if there is a back arc, we must have a cycle.

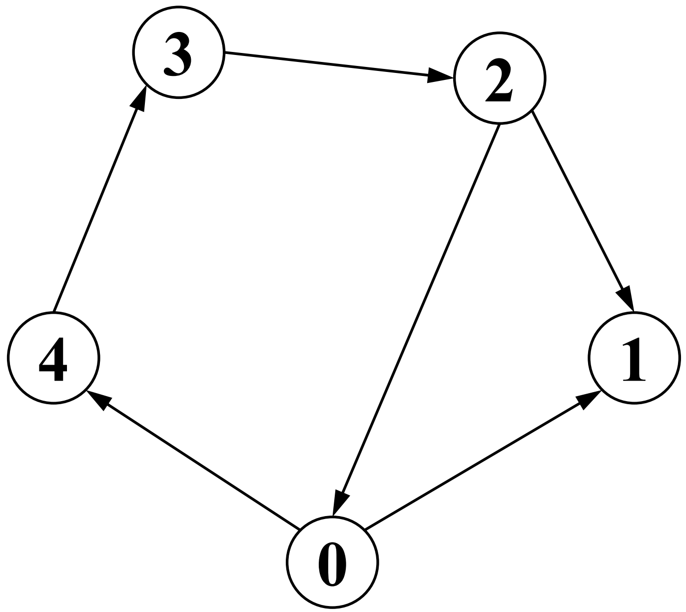
- Suppose that DFS is run on a digraph G. Then G is acyclic if and only if G does not contain a back arc.

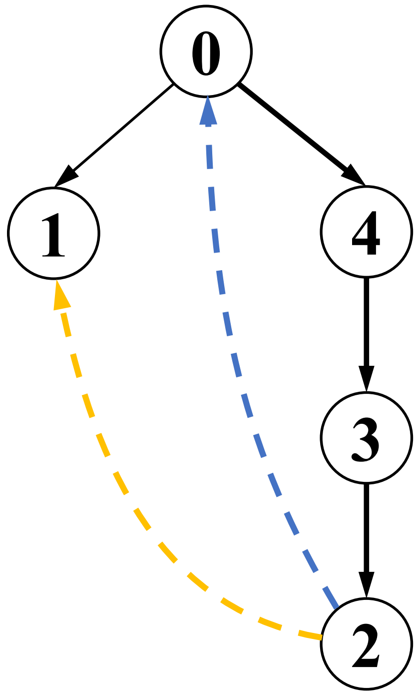
Example: Cycles in Graphs

- A graph can have several cycles



Using DFS to Find Cycles in Digraphs





Back arc detected!

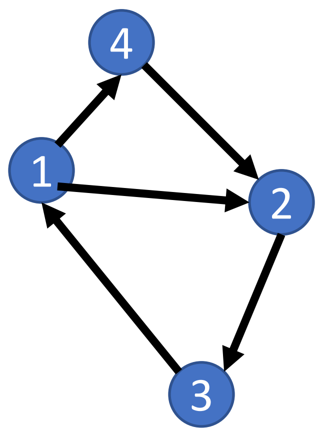
Once DFS finds a cycle, the stack contains the nodes that form the cycle.

Girth and Digirth

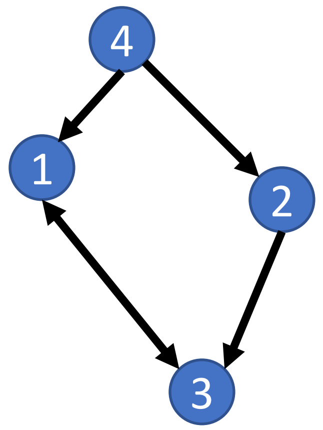
- **Definition.** For a graph (with a cycle), the length of the shortest cycle is called the girth of the graph. If the graph has no cycle then the girth is undefined but may be viewed as $+\infty$.

- **Convention.** For a digraph we use the term girth for its **underlying graph** and the term directed girth for the length of the smallest directed cycle.

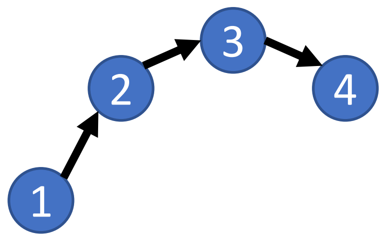
Examples: Girth of a Graph (Digraph)



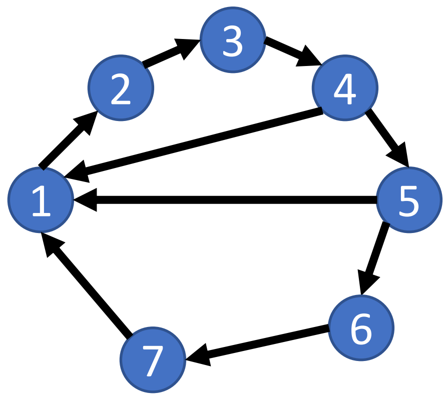
Girth: 3
Directed girth: 3



Girth: 4
Directed girth: 2

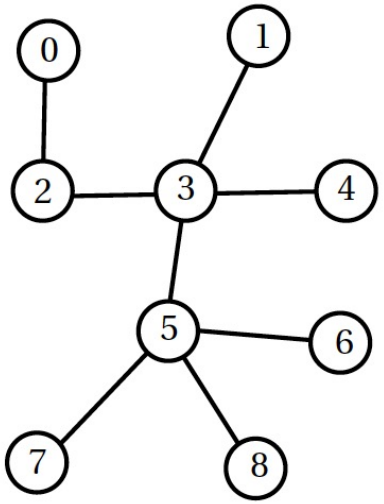


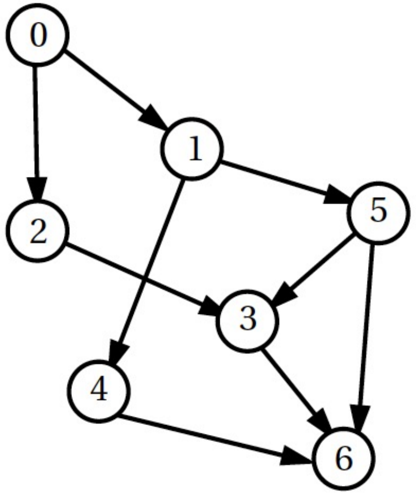
Girth: $+\infty$
Directed girth: $+\infty$

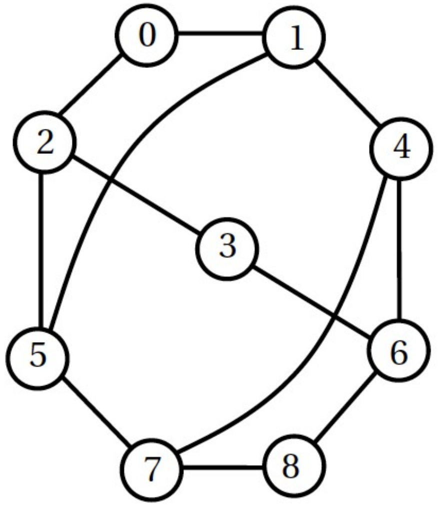


Girth: 3
Directed girth: 4

Examples: Girth of a Graph (Digraph)







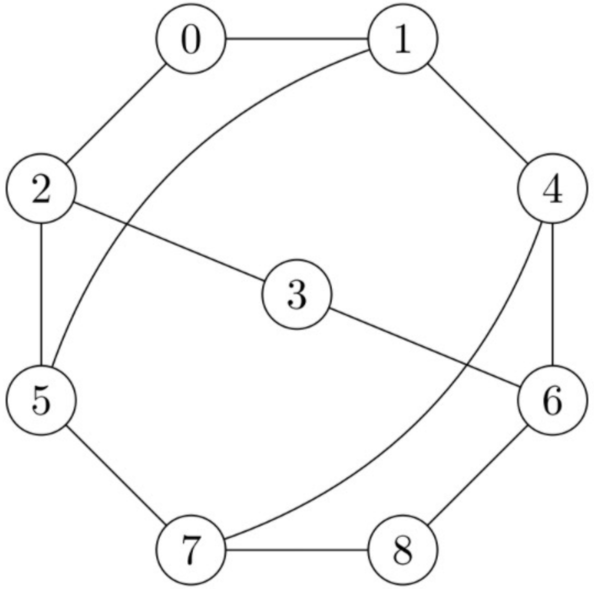
Girth: $+\infty$
Directed girth: undefined

Girth: 3
Directed girth: $+\infty$

Girth: 4
Directed girth: undefined

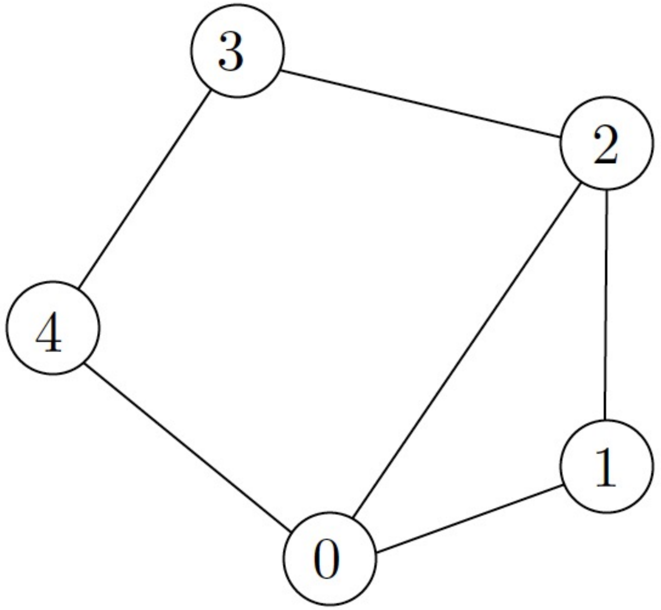
Finding the Girth of a Graph

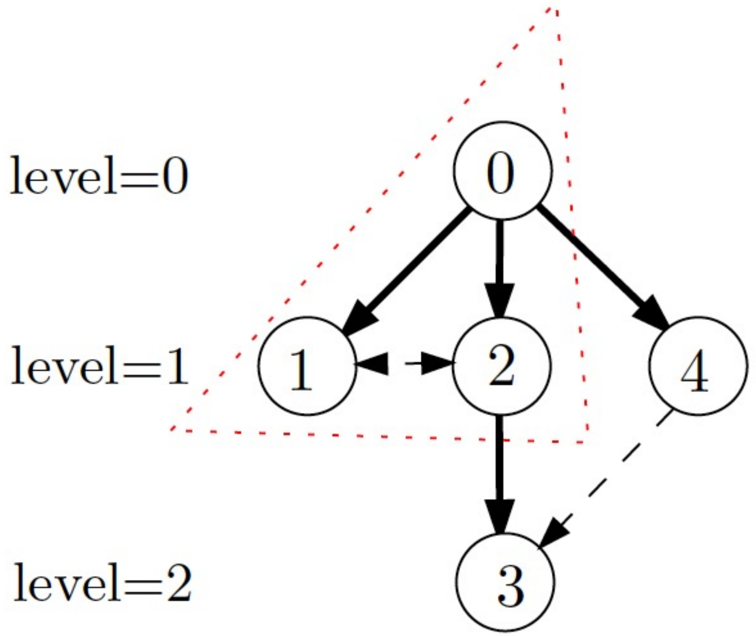
- An easy-to-implement **DFS** idea may not work properly. Consider the **DFS** tree originating from vertex 0 of the graph below. Which is the smallest cycle it finds with the back edges? Which smaller cycle does it miss?



Finding the Girth of a Graph

- **Using BFS to find cycles in graphs**. Cycles can also be easily detected in a graph using BFS. Finding a cycle of minimum length in a graph is not difficult using BFS (better than DFS).





Finding the Girth of a Graph

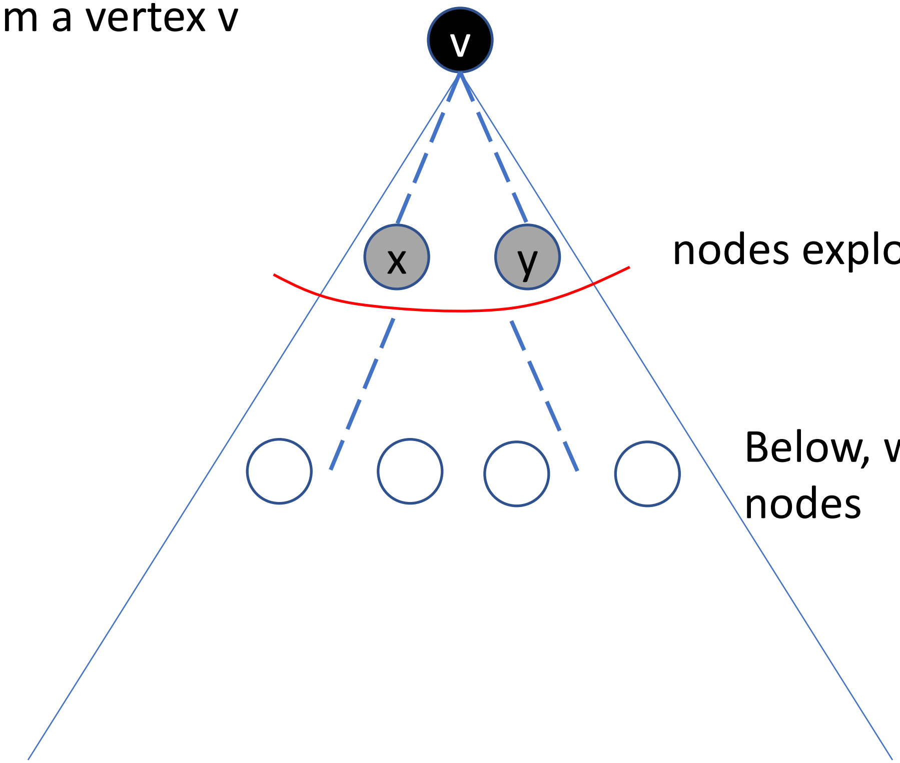
- Perform BFSVisit() $|V|$ times, starting at each vertex $\mathrm{v}' \in V$ in turn.

- If during a BFSVisit, we encounter a grey neighbour (rather than a white) we have found a cycle
  - The grey node was visited before following another path

- An important property of BFS is that if it runs from vertex v, then every vertex s, when it is first reached, then the path that was found from v to s is minimal. Thus, reaching v' from v with BFS finds the shortest path from v to v', namely the shortest cycle that contains v.

Example: Girth of a Graph

Executing BFSVisit from a vertex v

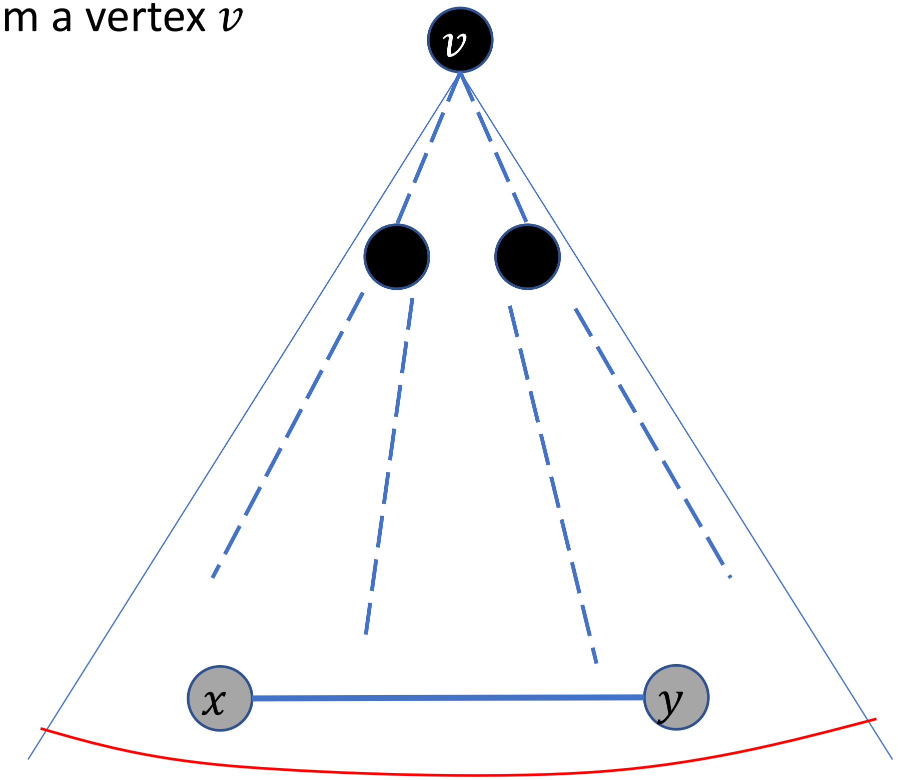


nodes explored level by level

Below, we should only find white nodes

Example: Girth of a Graph

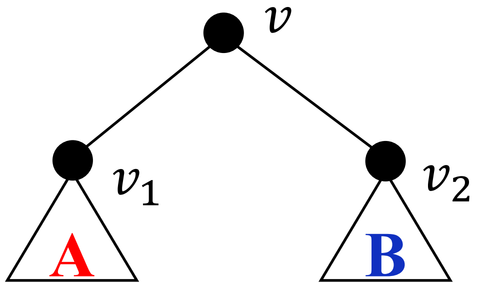
Executing BFSVisit from a vertex v

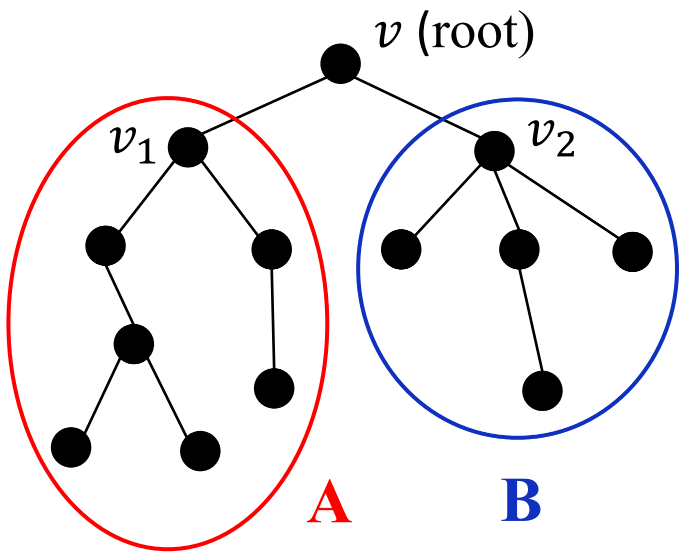


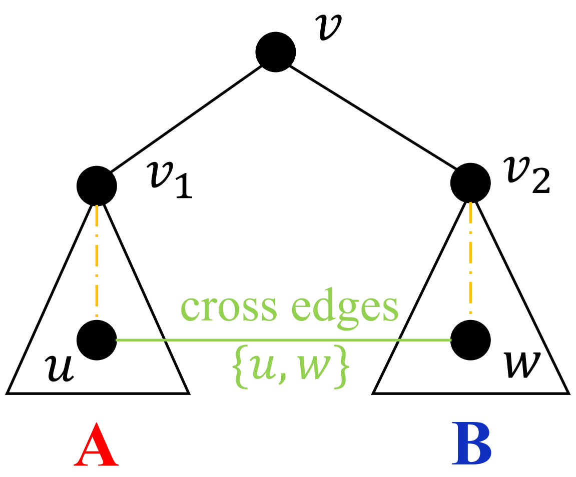
If a cross edge/arc exits then it will connect to a grey vertex

Both x and y are reachable from v therefore there is a cycle through the cross edge

Example: Girth of a Graph



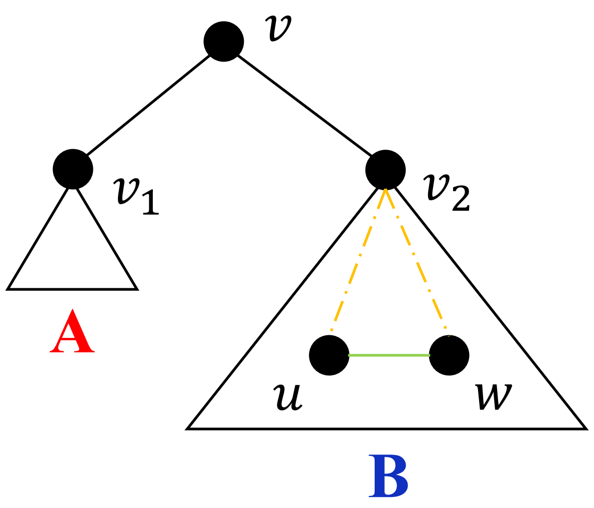




Distance from root v

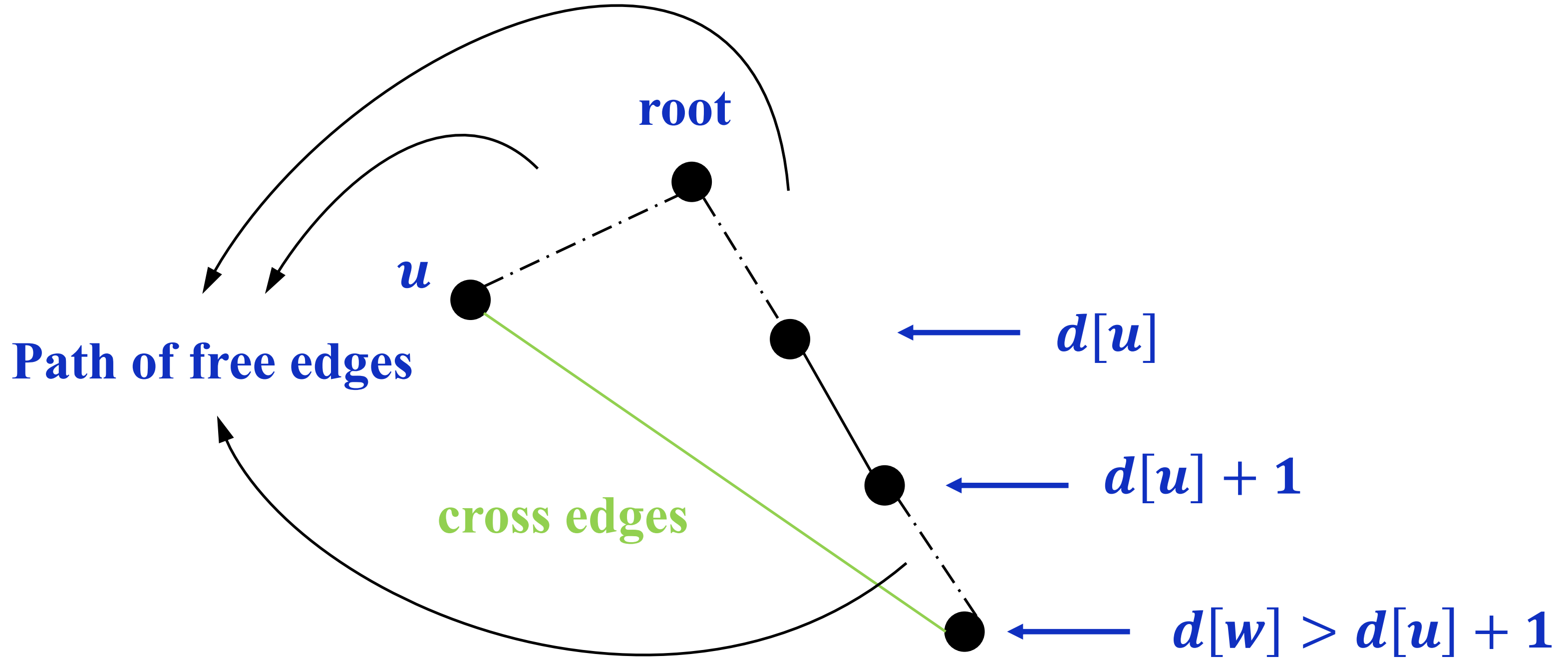
$d[u] + d[w] + 1$

(length of cycle containing v)



BFS: Cross Edges in a Graph

- Let $\{u, w\}$ be a cross edge in a tree. Then either $d[u] = d[w]$ or $|d[u] - d[w]| = 1$
- **Sketch of proof:** Suppose $|d[u] - d[w]|$ >1



$d[w]$ > distance from root to w (shortest path from root to w contains the cross edge)
***contradiction**

Facts about Cycle Length

- If vertex v is on at least one cycle then BFS starting at v will find it.
- On detection of a cross edge between descendants u and w, determine whether u and w are in different subtrees below v (the root of the tree).
  - If yes, then a cycle of length $d[u]$+$d[w]$+1 is found.
  - If no, then a cycle of shorter length is found (but avoids v).
- If $d[u]$=$d[w]$ then odd length, where v is a common ancestor.
- Otherwise, $d[u]$+1=$d[w]$ and even length.

Facts about Cycle Length

- cross edge $\{u,w\}$

1. $d[u] = d[w]$

cycle length: $2d[u] + 1 \Longrightarrow$ odd length

cross edge

2. $d[u] + 1 = d[w]$

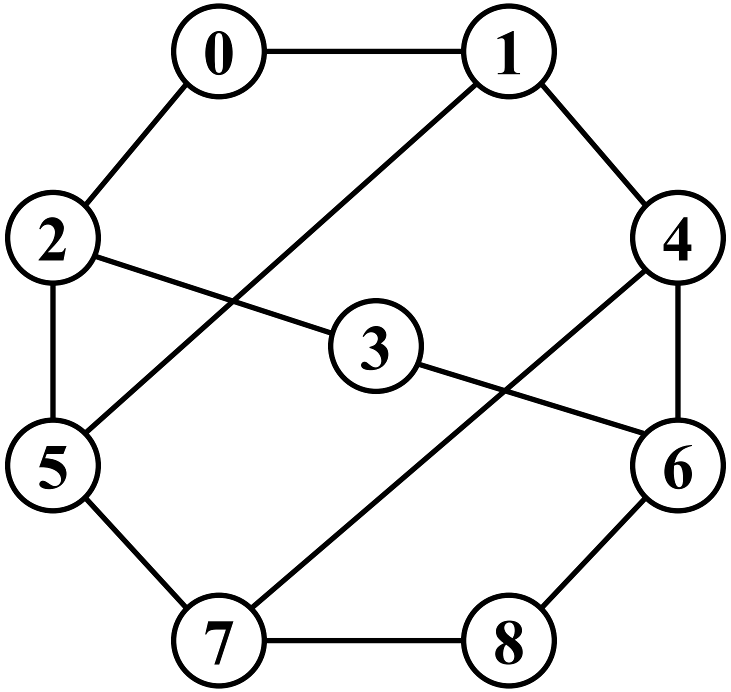
cycle length: $d[u] + d[w] + 1$
$= d[u] + d[u] + 1 + 1$
$= 2d[u] + 2 \Longrightarrow$ even length

Finding the Girth of a Graph

- To compute girth, we perform BFSVisit(v) procedure once for each $v \in V(G)$ and take minimum.
  - If a grey neighbour is met, e.g., an edge (x, y) is explored from x where y is grey, continue to the end of the current level and then stop.
  - For each edge (x, y) as above on this level, if v is the lowest common ancestor of x and y in the BFS tree, then there is a cycle containing x, y, v of length $l = d(x) + d(y) + 1$.
  - Report the minimum value of l obtained along the current level.

- The minimum of these lengths at the level is the smallest cycle that involves v
- The smallest cycle among all possible start vertices v is the girth.

Example (1): Finding the Girth of a Graph

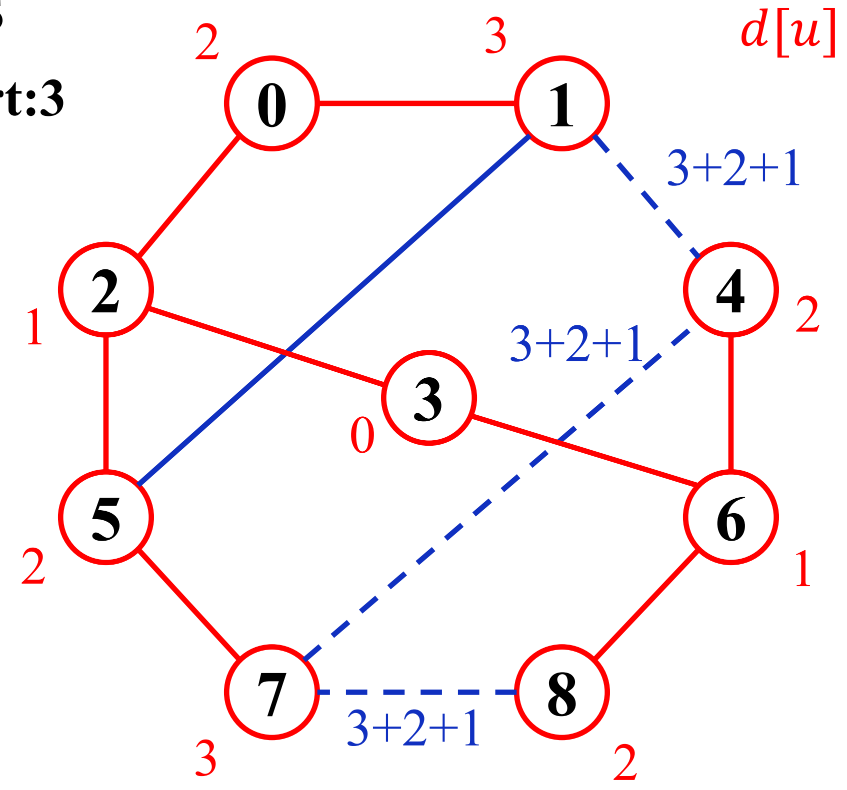


- Run BFS for 3, shortest cycle has length 6.
- Run BFS for any other vertex, shortest cycle has length 4.
- Girth of the graph is 4..

Example (1)

BFS

Start:3

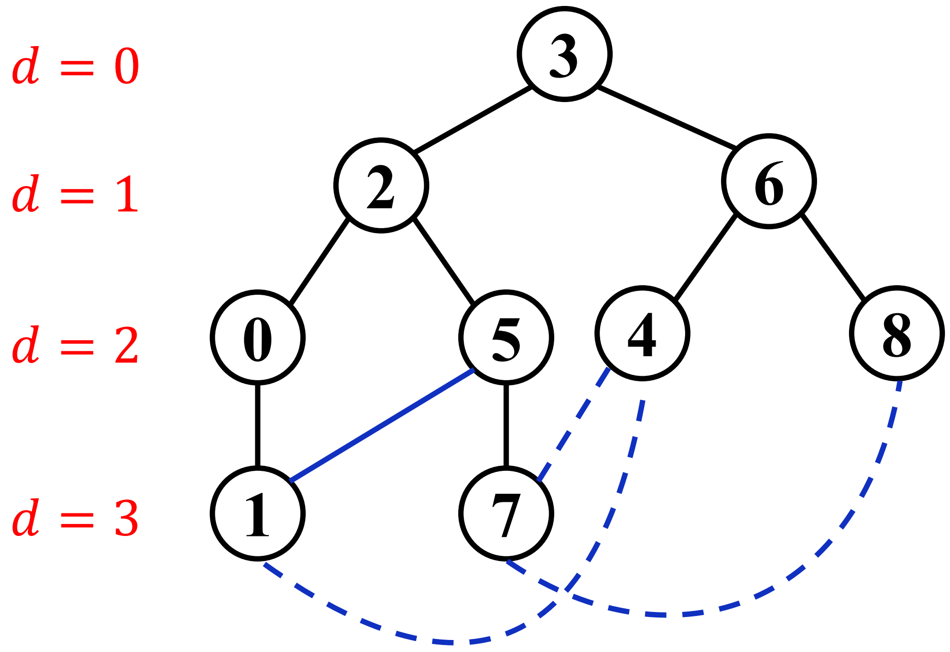


tree edges

cross edge same subtree

tree edges different subtree

(relative to root 3)



Shortest cycle containing 3 has length 6

Example (1)

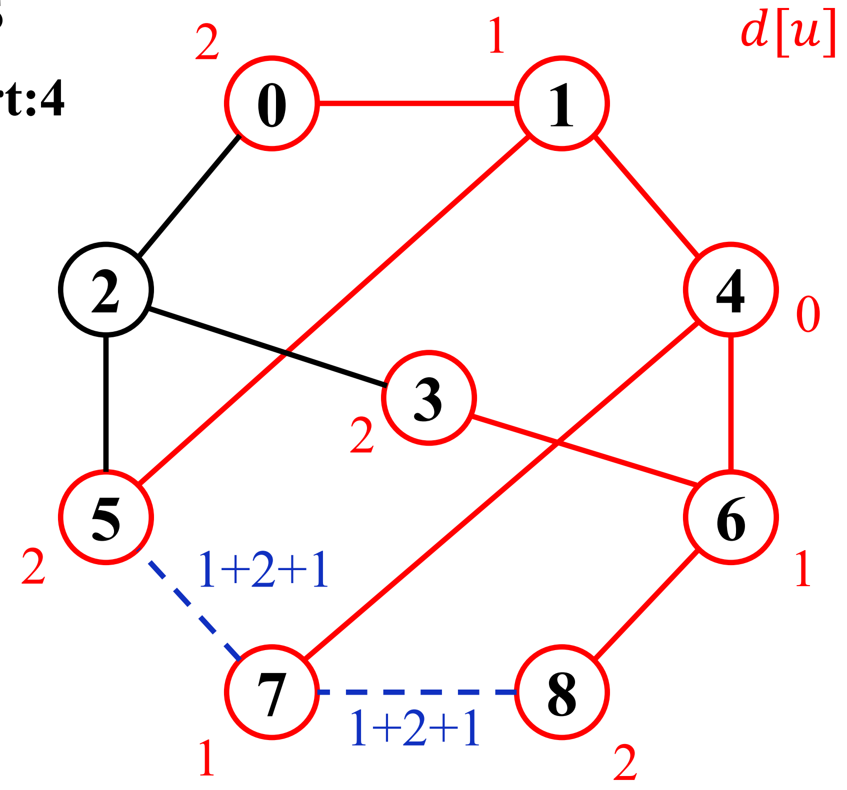
tree edges

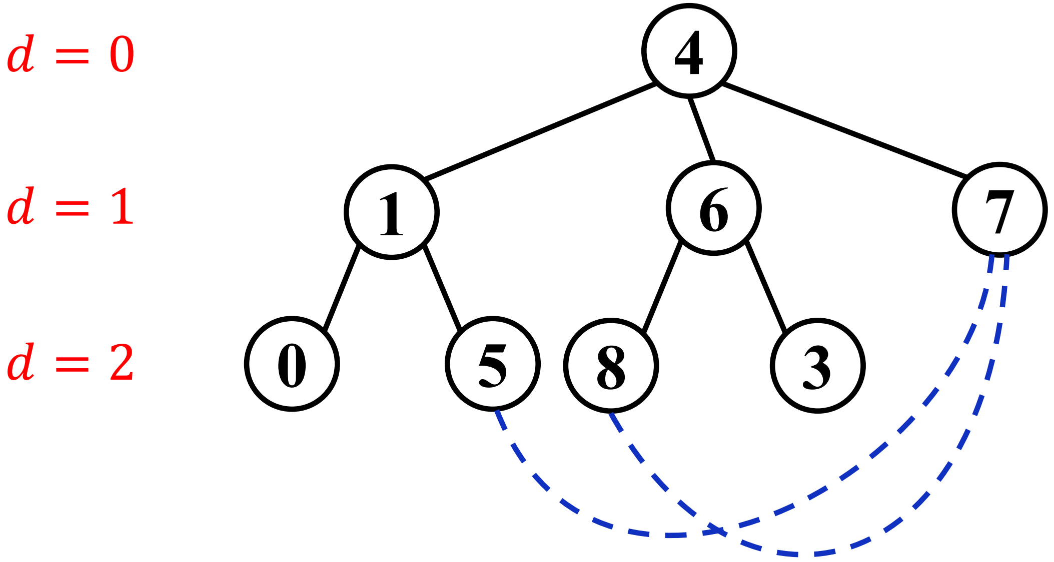
cross edge same subtree

tree edges different subtree (relative to root 4)

BFS

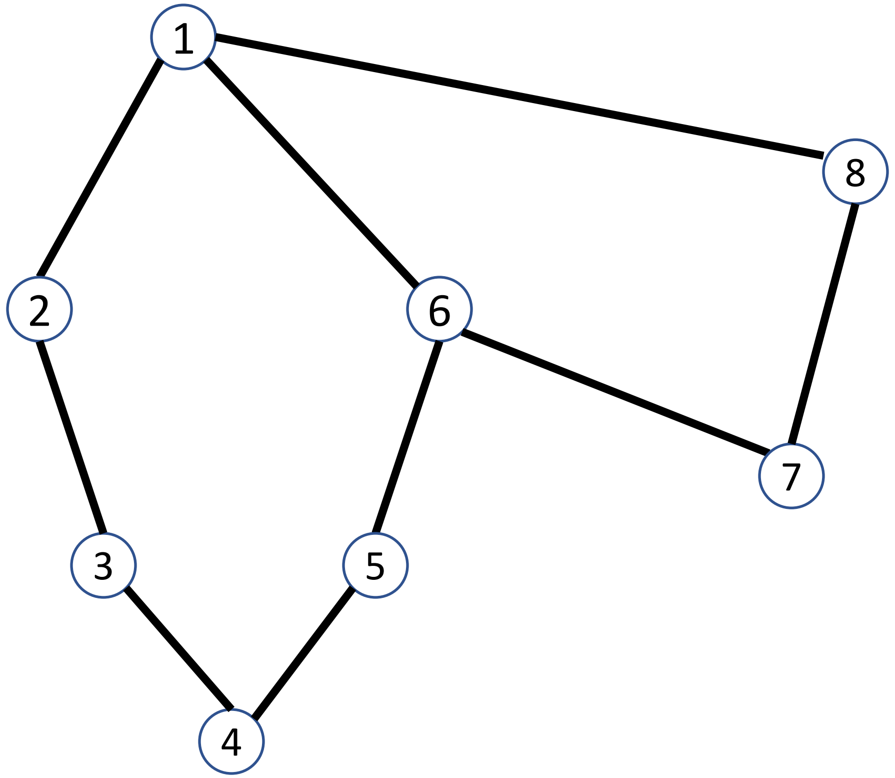
Start:4



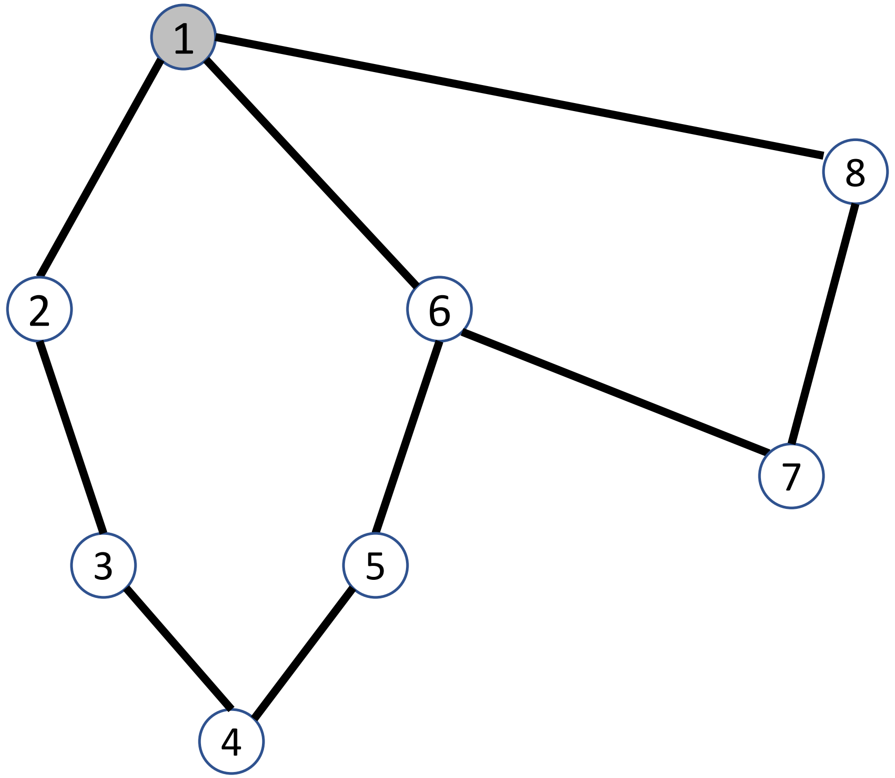


Shortest cycle containing 4 has length 4

Exercise: Finding the Smallest Cycle involving Vertex 1

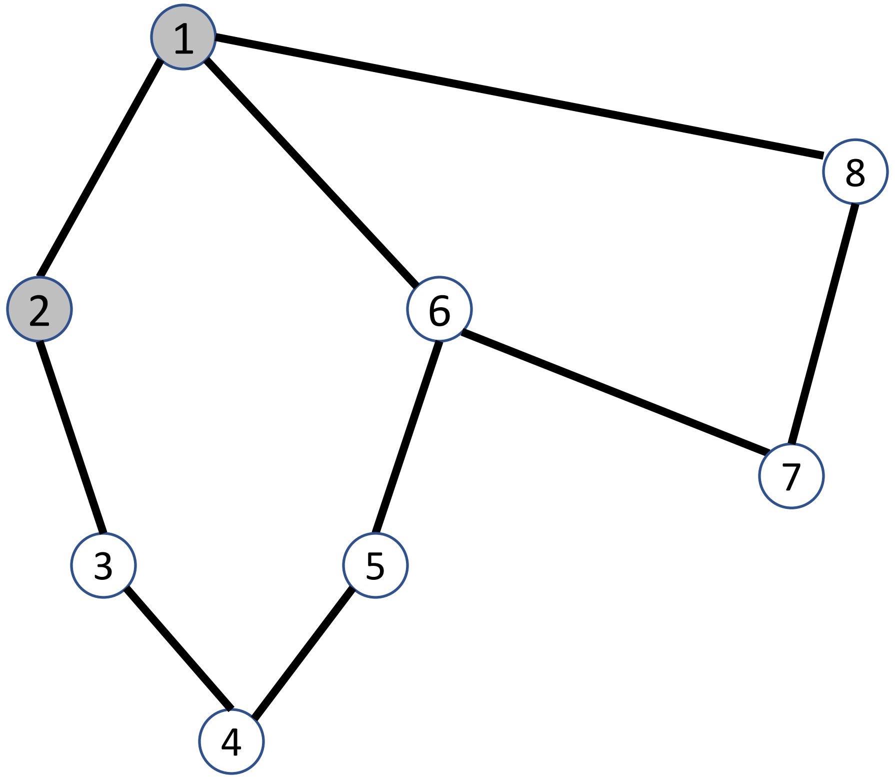


Exercise: Finding the Smallest Cycle involving Vertex 1



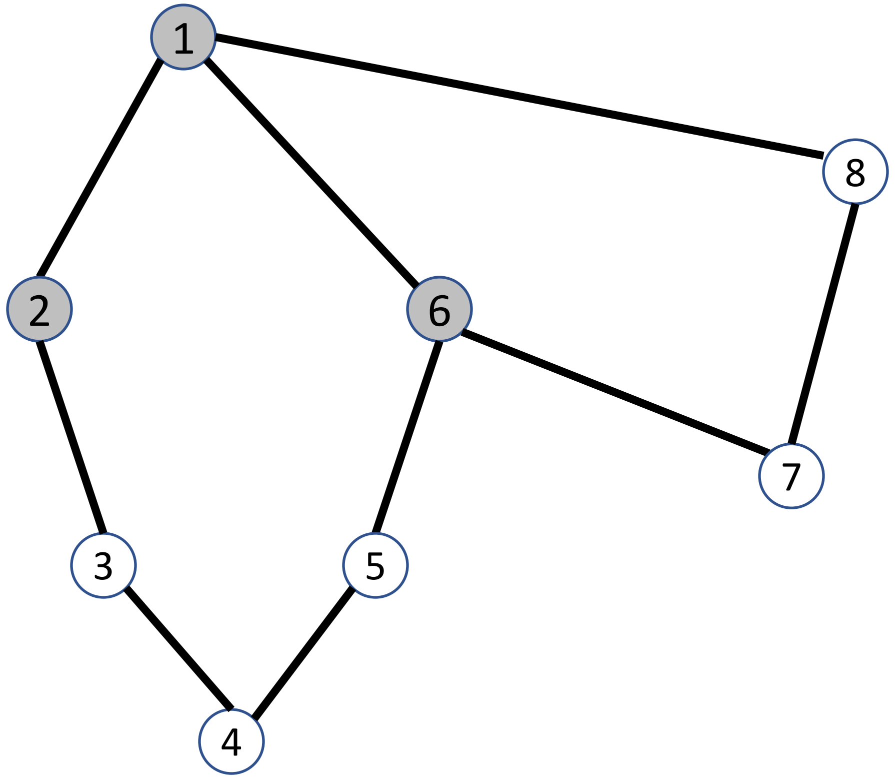
Queue: 1

Exercise : Finding the Smallest Cycle involving Vertex 1



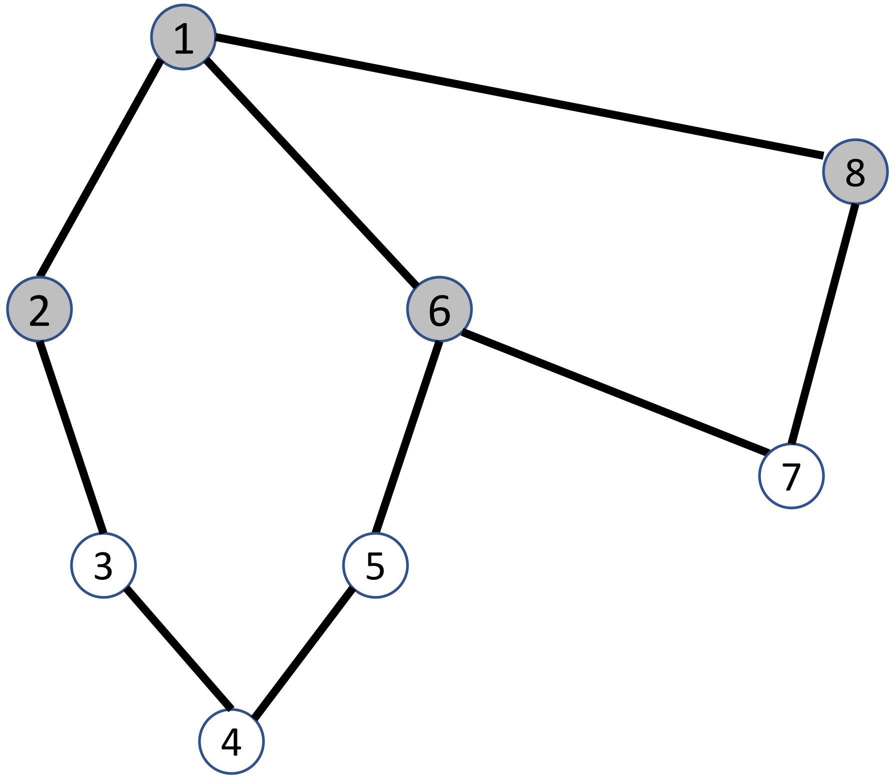
Queue: 1 2

Exercise: Finding the Smallest Cycle involving Vertex 1



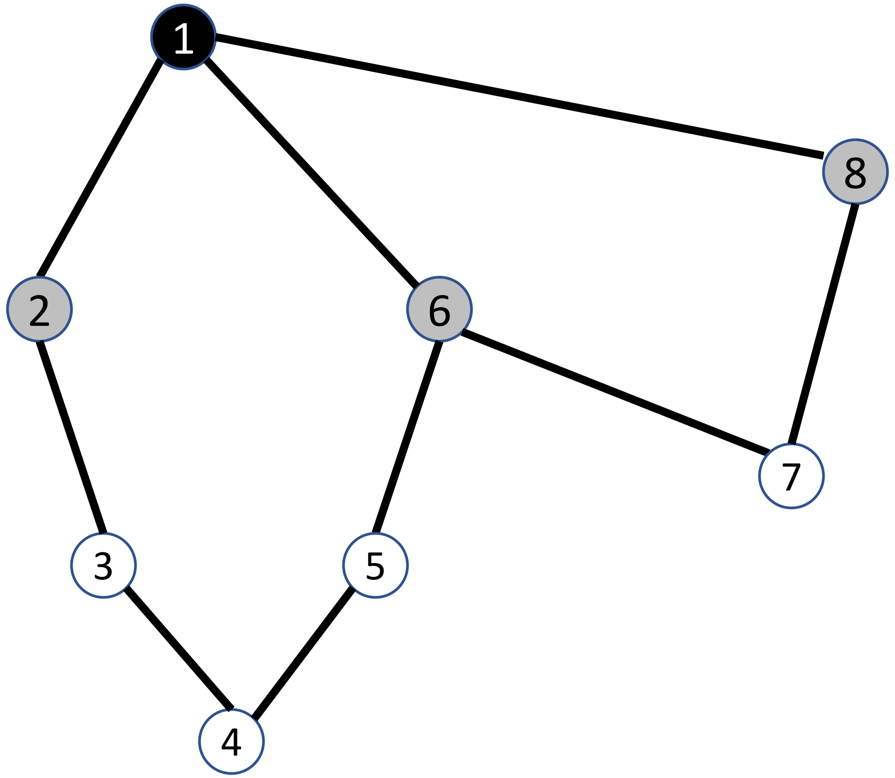
Queue: 1 2 6

Exercise: Finding the Smallest Cycle involving Vertex 1



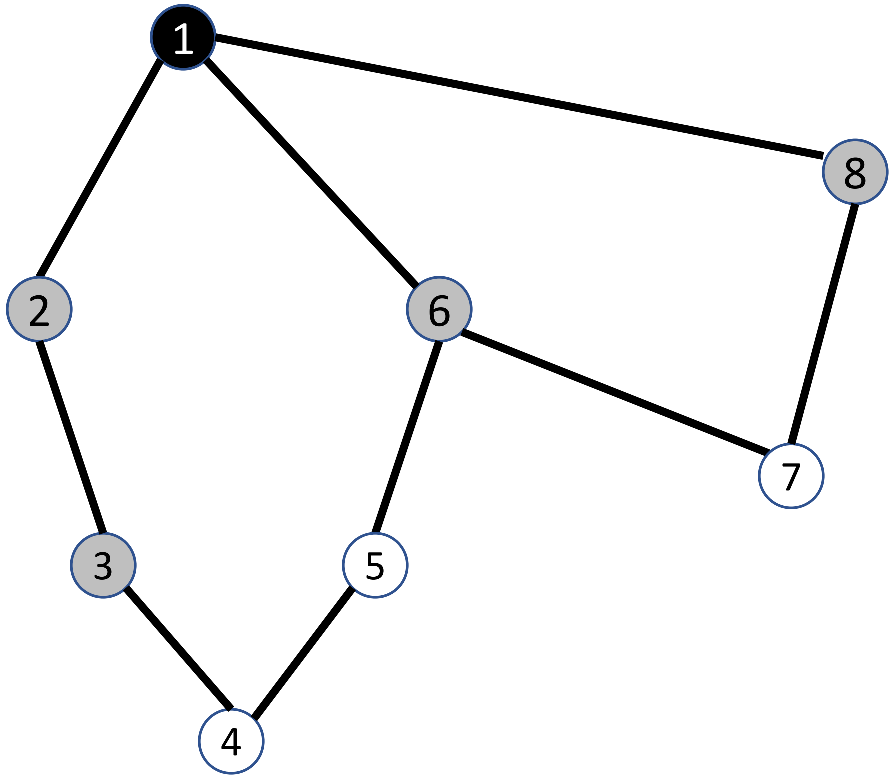
Queue: 1 2 6 8

Exercise: Finding the Smallest Cycle involving Vertex 1



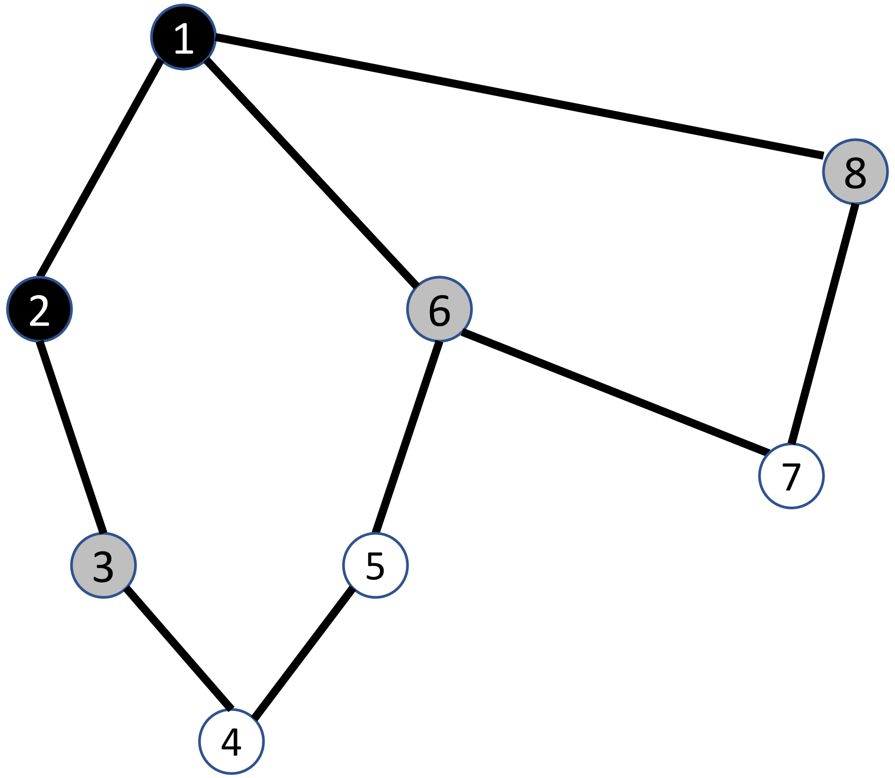
Queue: 2 6 8

Exercise: Finding the Smallest Cycle involving Vertex 1



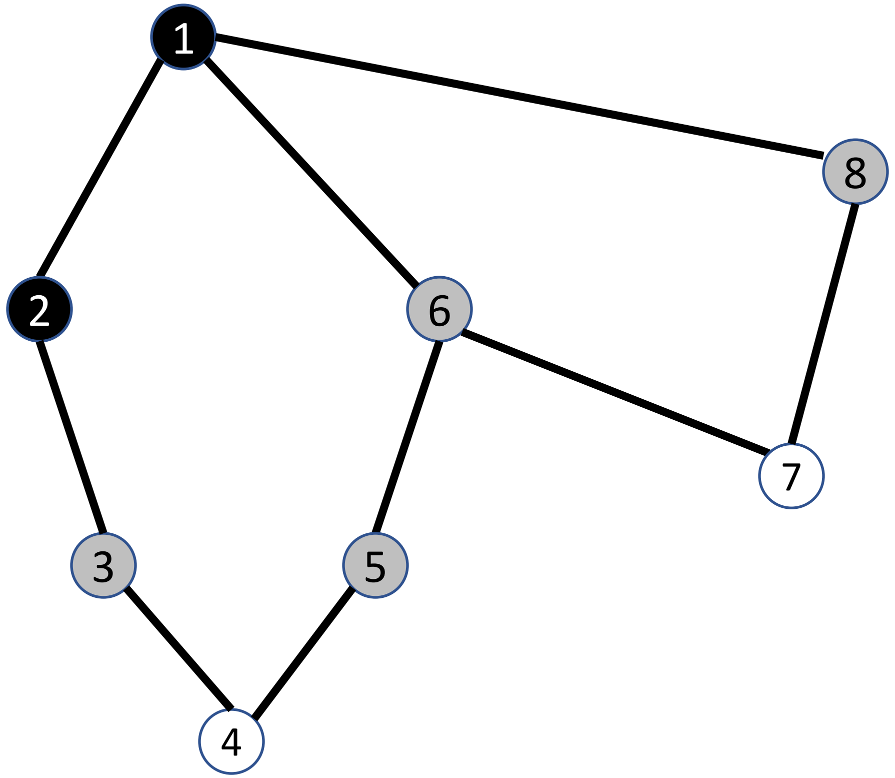
Queue: 2 6 8 3

Exercise: Finding the Smallest Cycle involving Vertex 1



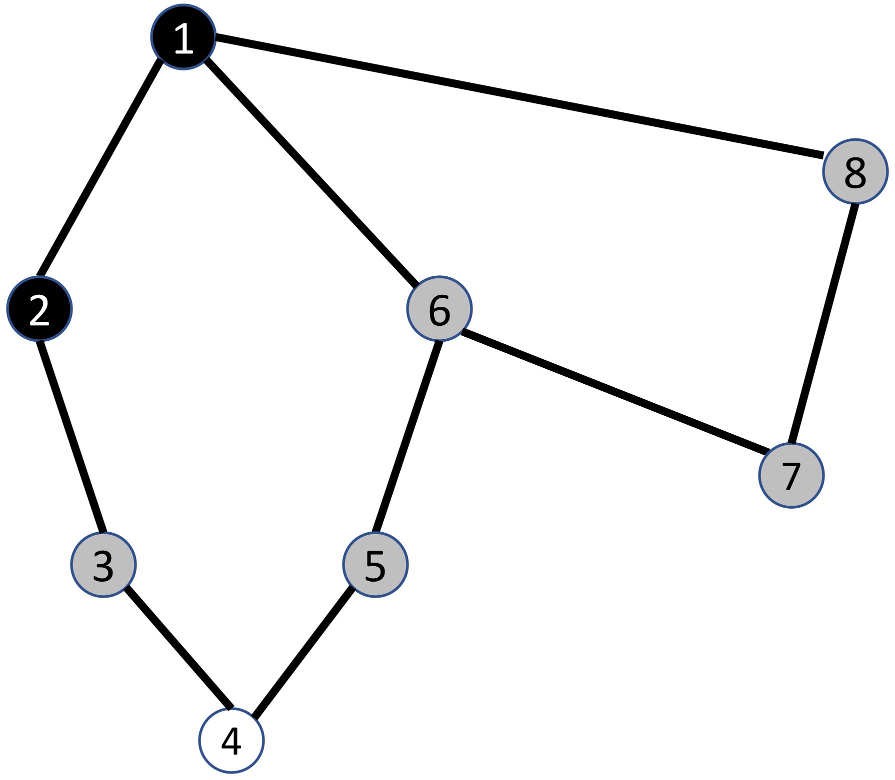
Queue: 6 8 3

Exercise: Finding the Smallest Cycle involving Vertex 1



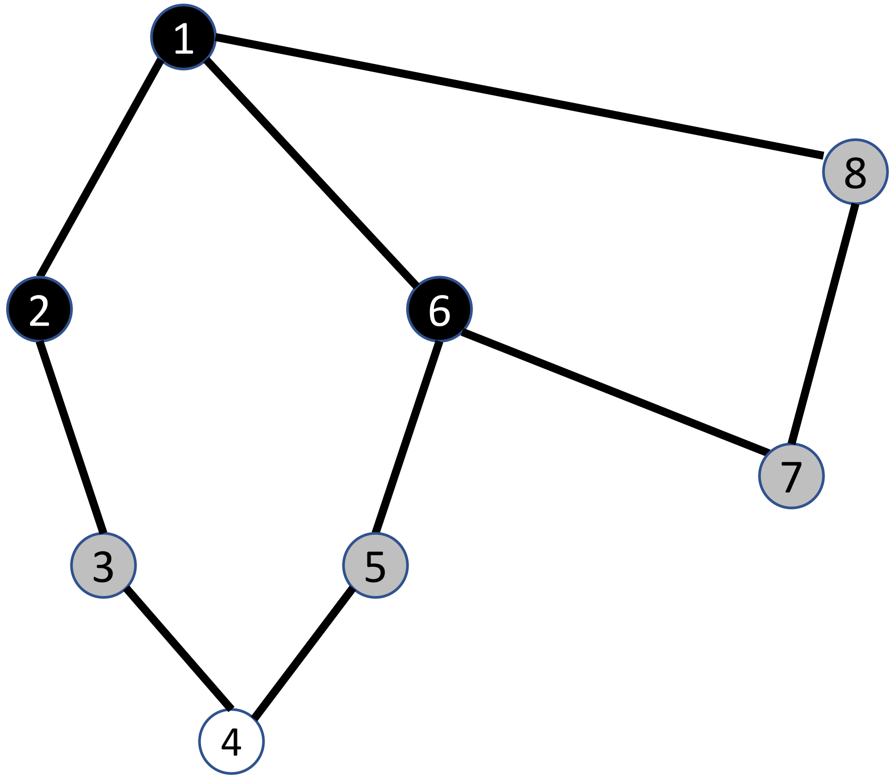
Queue: 6 8 3 5

Exercise: Finding the Smallest Cycle involving Vertex 1



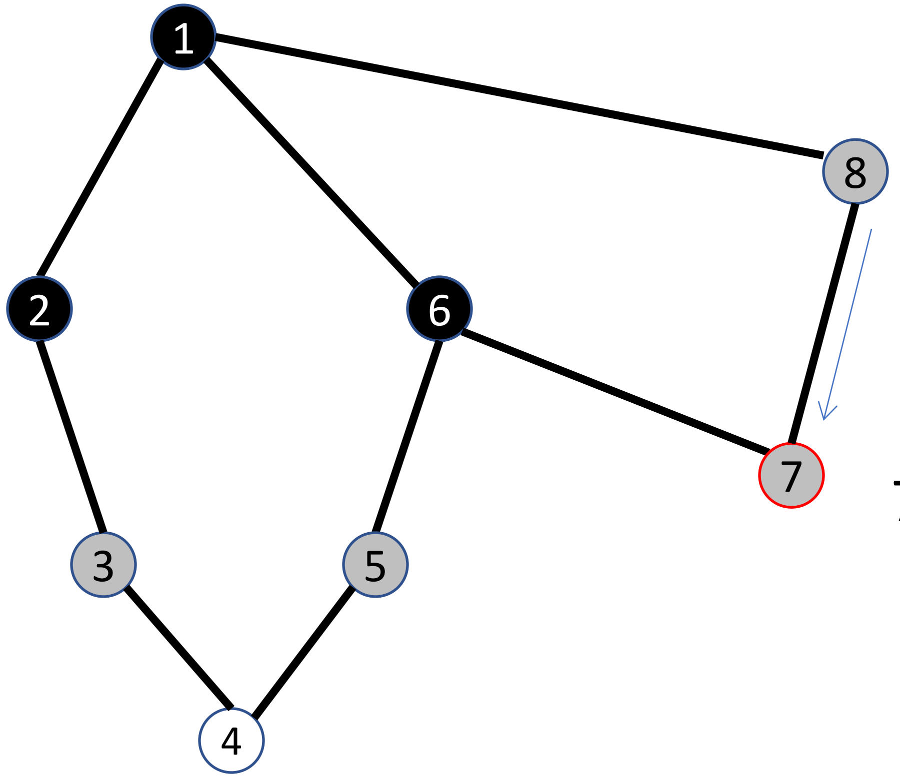
Queue: 6 8 3 5 7

Exercise: Finding the Smallest Cycle involving Vertex 1



Queue: 8 3 5 7

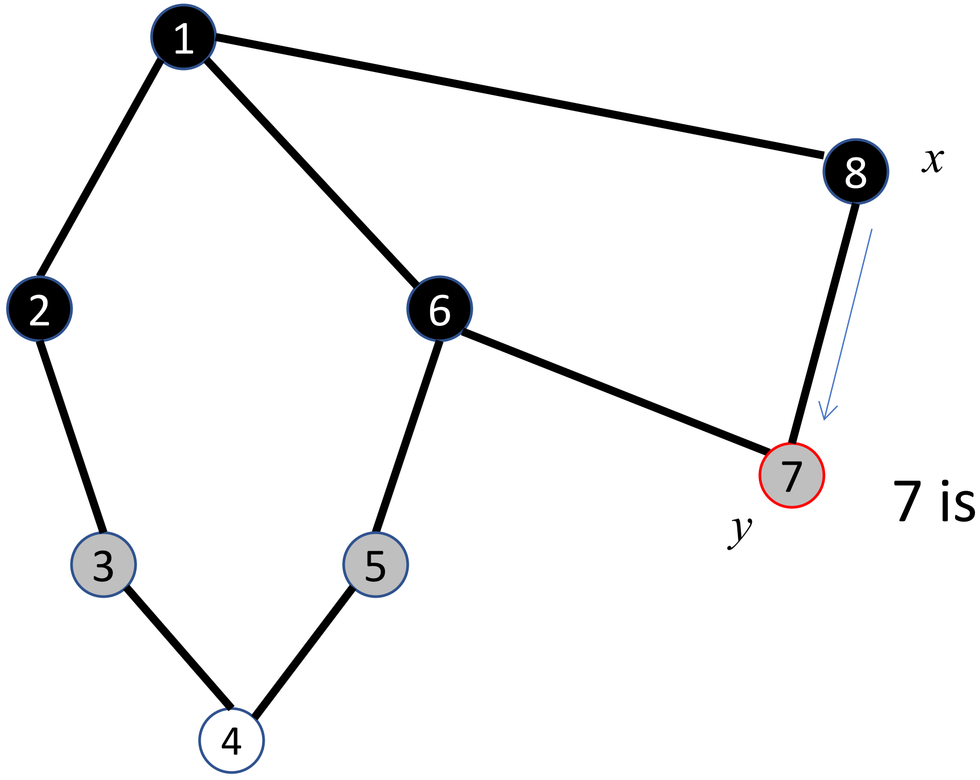
Exercise: Finding the Smallest Cycle involving Vertex 1



7 is already grey!

Queue: 8 3 5 7

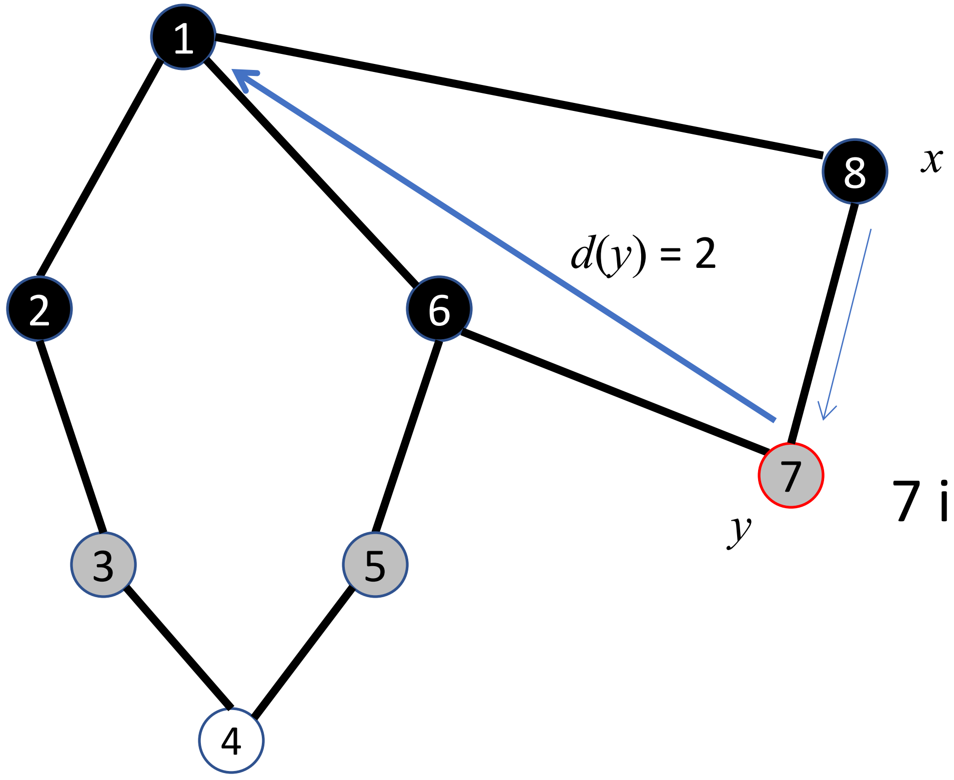
Exercise: Finding the Smallest Cycle involving Vertex 1



7 is already grey!

Queue: 3 5 7 (stopped)

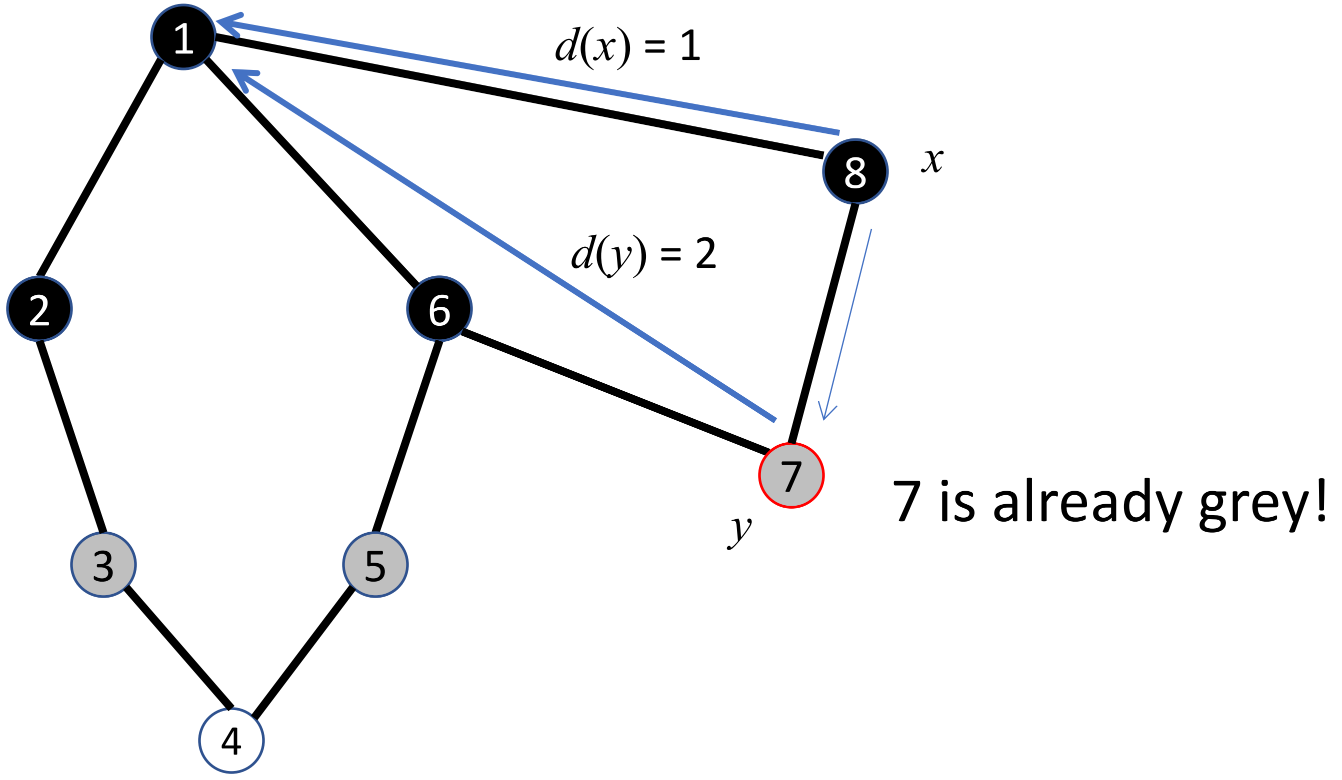
Exercise: Finding the Smallest Cycle involving Vertex 1



7 is already grey!

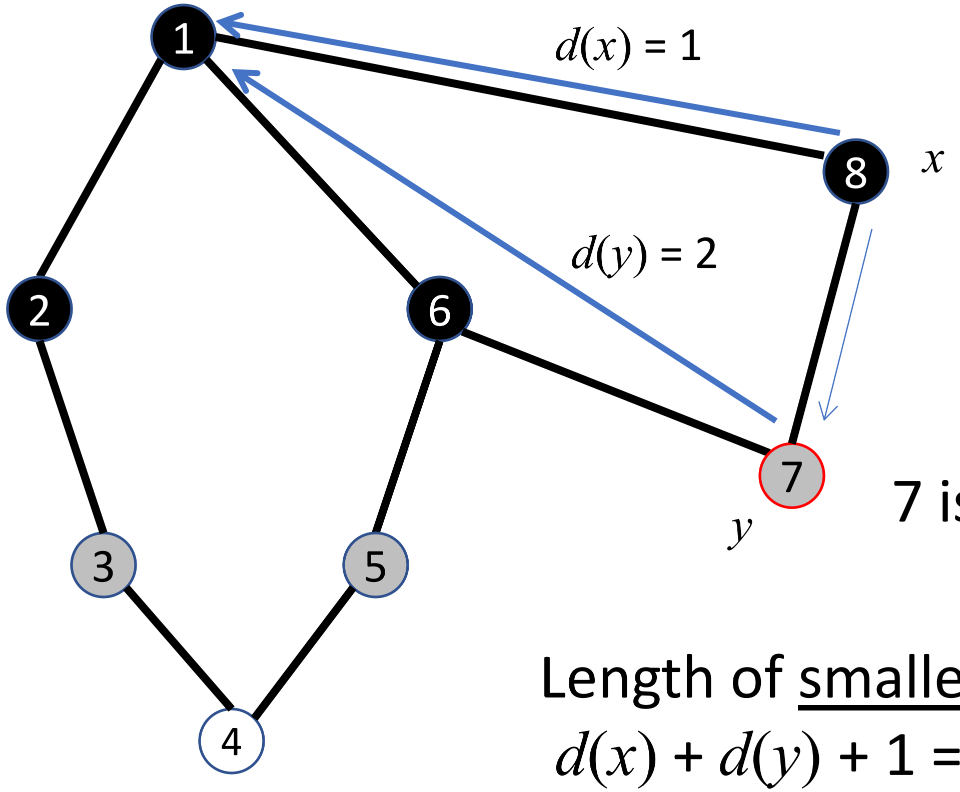
Queue: 3 5 7 (stopped)

Exercise: Finding the Smallest Cycle involving Vertex 1



Queue: 3 5 7 (stopped)

Exercise: Finding the Smallest Cycle involving Vertex 1



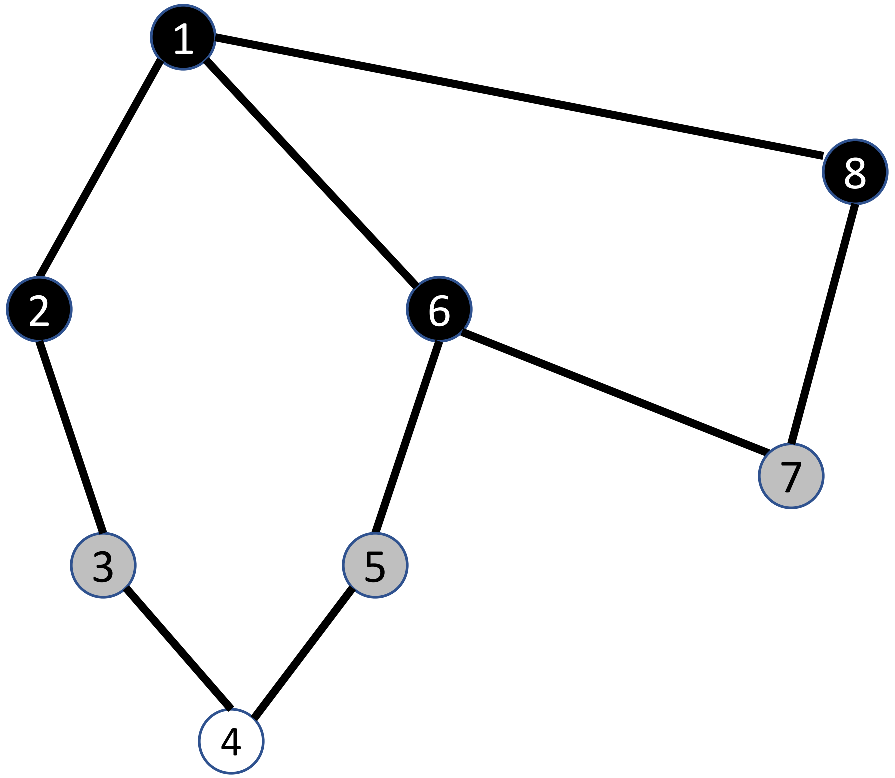
7 is already grey!

Length of <u>smallest</u> cycle with vertex 1

$d(x) + d(y) + 1 = 4$

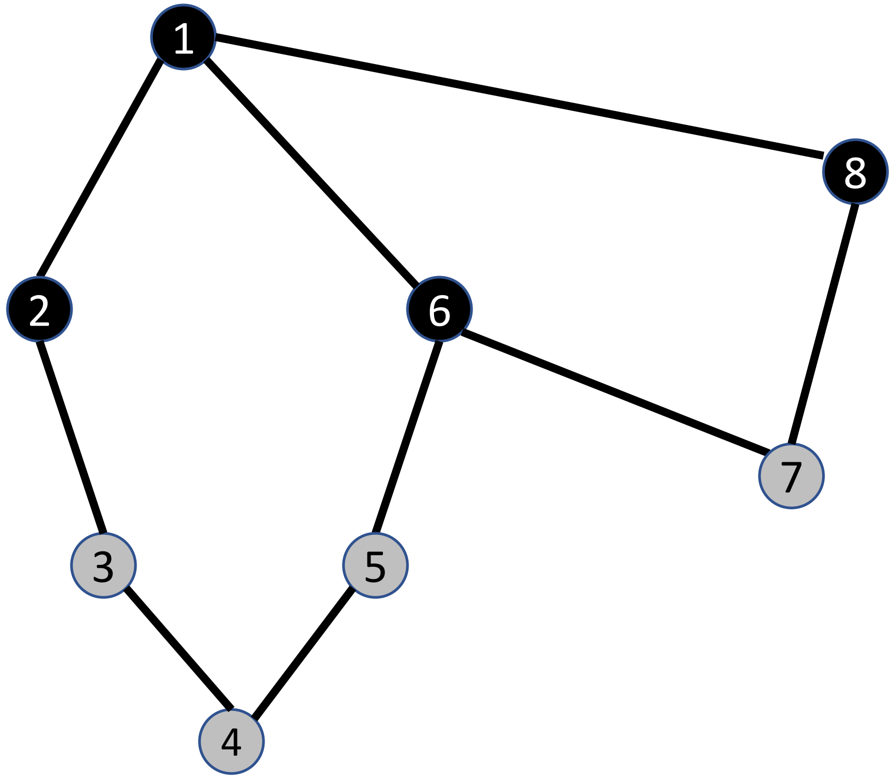
Queue: 3 5 7 (stopped)

Exercise: Finding the Smallest Cycle involving Vertex 1



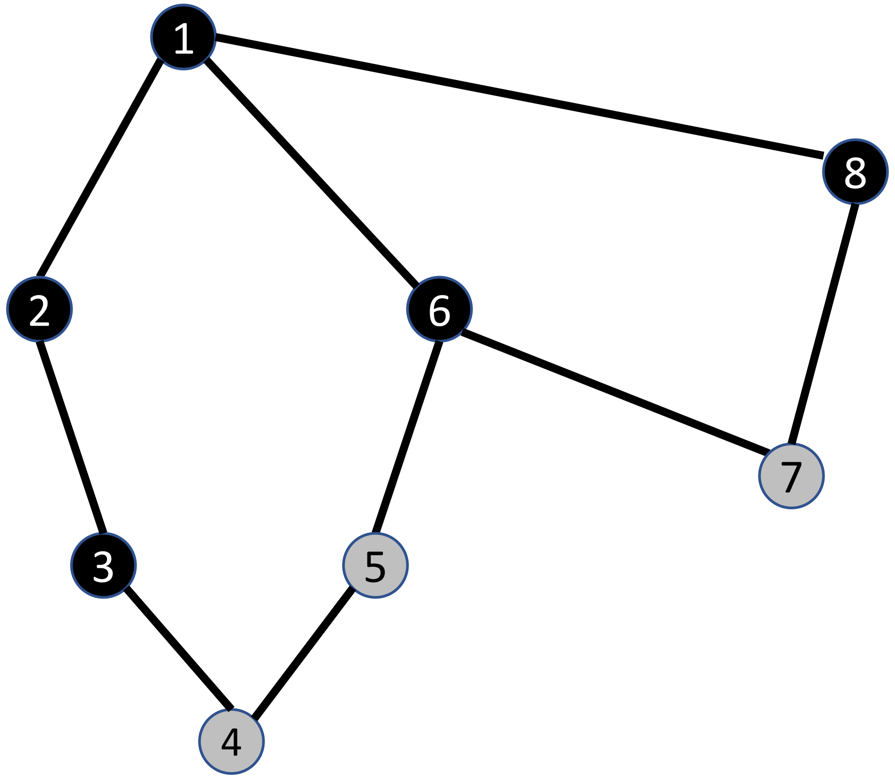
Queue: 3 5 7

Exercise: Finding the Smallest Cycle involving Vertex 1



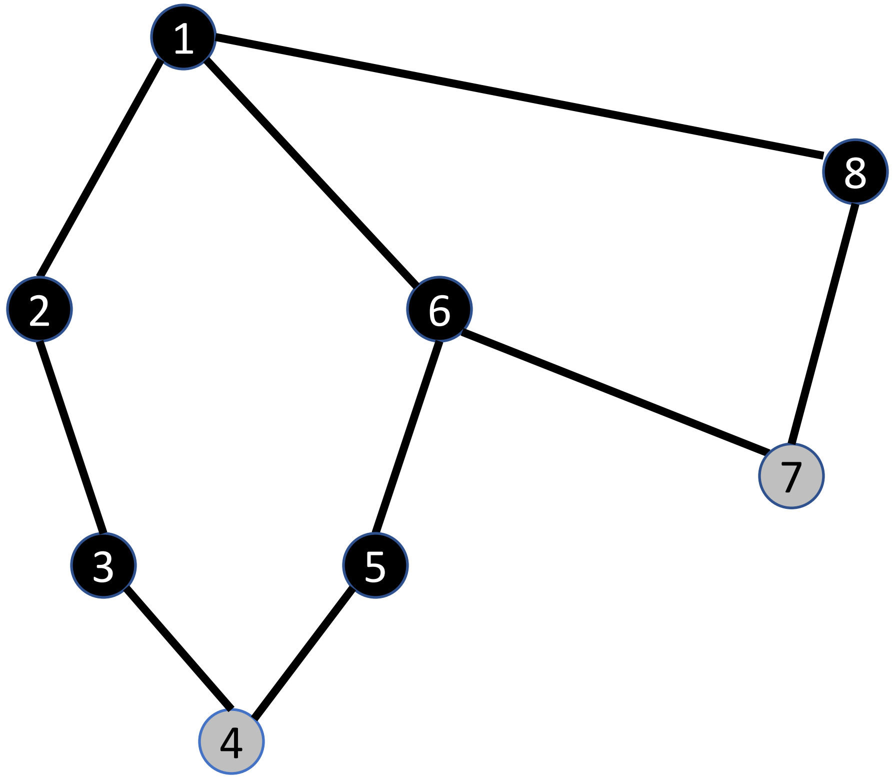
Queue: 3 5 7 4

Exercise: Finding the Smallest Cycle involving Vertex 1



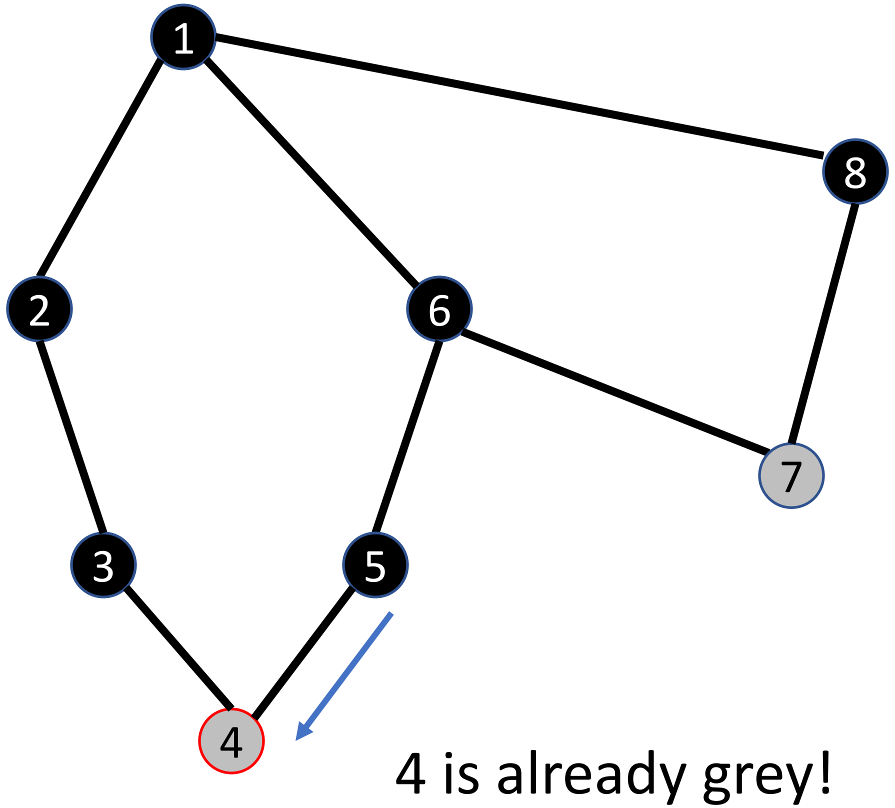
Queue: 5 7 4

Exercise: Finding the Smallest Cycle involving Vertex 1



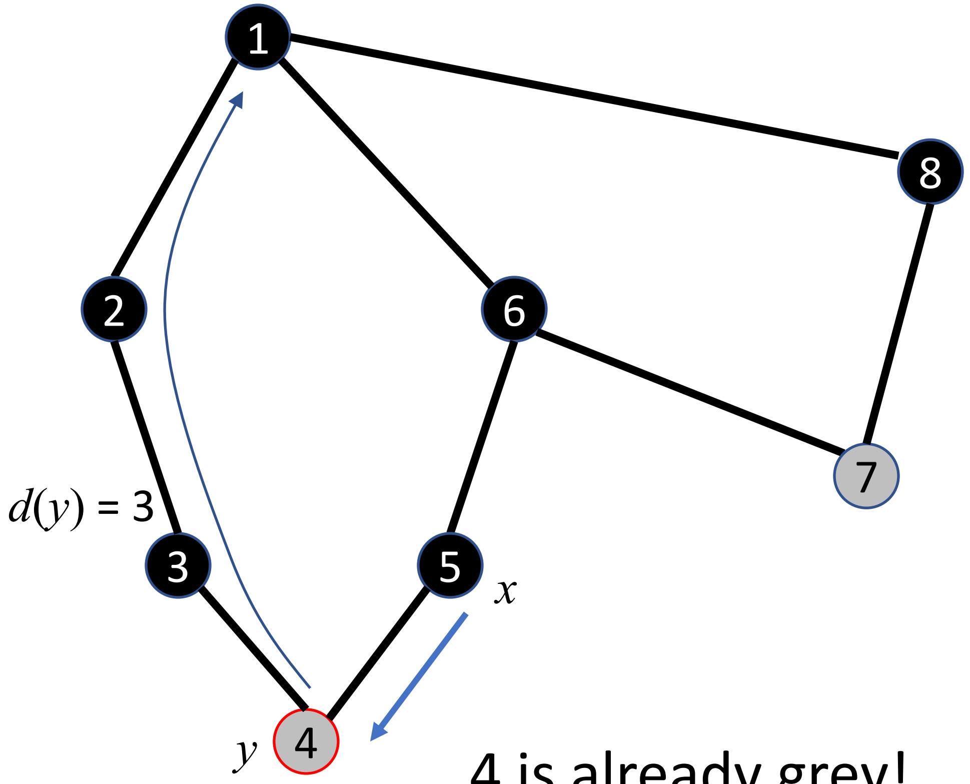
Queue: 7 4

Exercise: Finding the Smallest Cycle involving Vertex 1



Queue: 7 4 (stopped)

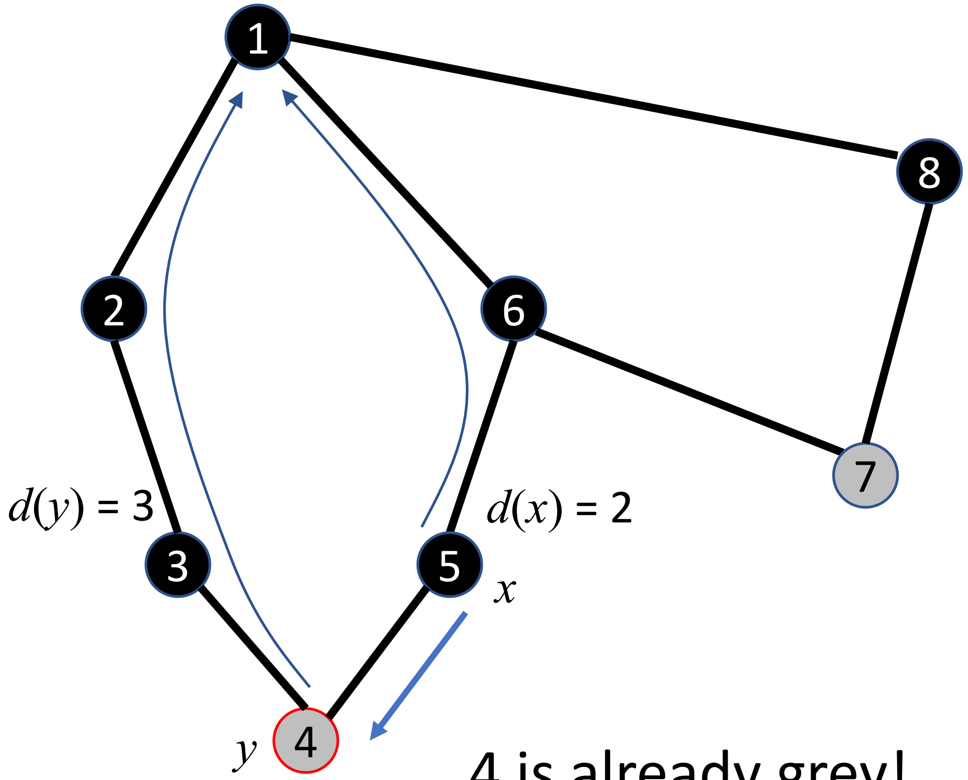
Exercise: Finding the Smallest Cycle involving Vertex 1



4 is already grey!

Queue: 7 4 (stopped)

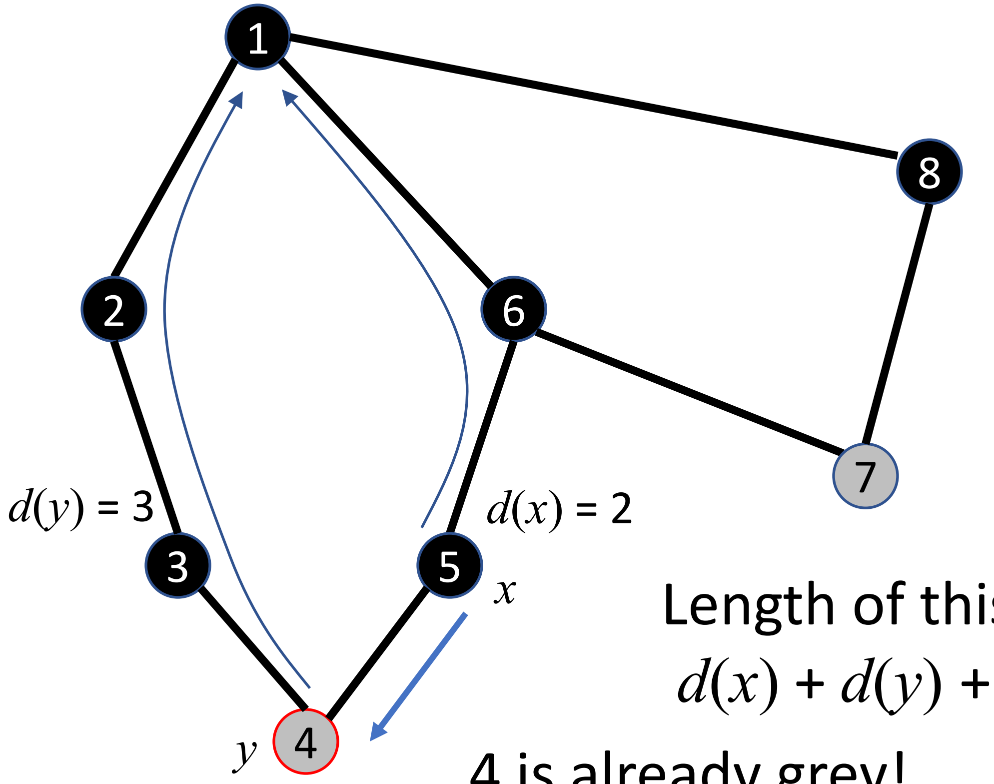
Exercise: Finding the Smallest Cycle involving Vertex 1



4 is already grey!

Queue: 7 4 (stopped)

Exercise: Finding the Smallest Cycle involving Vertex 1



Length of this cycle with vertex 1:
$d(x) + d(y) + 1 = 6$

4 is already grey!

Queue: 7 4 (stopped)

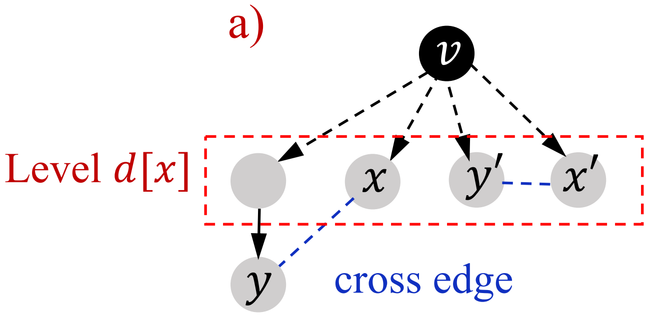
Correctness of the Algorithm

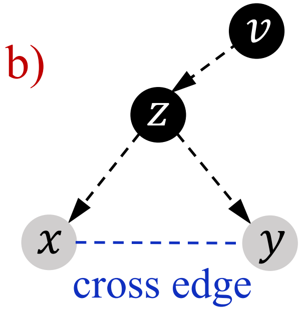
1. Meeting a grey neighbour means that we have found a cycle;
2. The cycle found in this way will be the shortest.

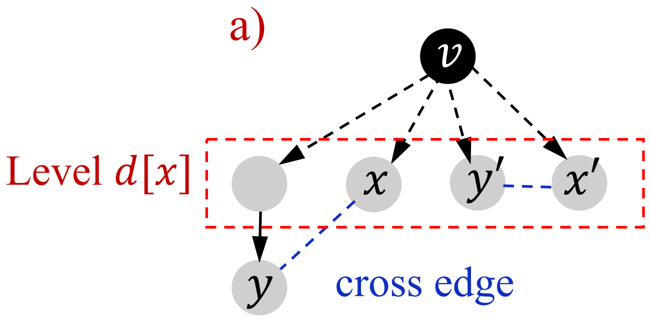
Running BFSVisit(v) above, we have two cases when we first meet a cross edge (x, y) and let $d[x] \leq d[y]$:

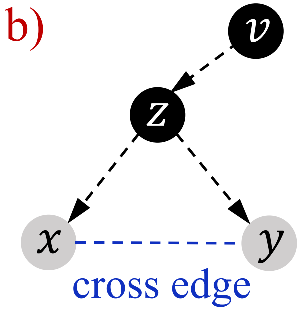
a) x and y are in different subtrees of v

b) x and y are in the same subtree of v









Running BFSVisit(v) above gives us two cases when we first meet a cross edge (x,y) and let $d[x] \le d[y]$:

a) x and y are in different subtrees of v: v,x,y are in the same cycle with length $d[x] + d[y] + 1 \le 2d[x] + 2$. All cross edges detected at the same level have the same upper bound of cycle length. Then, the algorithm finds the smallest cycle containing v.

b) x and y are in the same subtree of v: a smaller cycle (not including v) exists. Let z be the least common ancestor of x and y. When BFSVisit starts from z, we should find the smallest cycle containing z based on a).

Finding Directed Girth of Digraph

Following procedure finds the (length of the) shortest directed cycle through node v in a digraph.

1. Run BFSVisit(v) starting at node v.
2. The first time a back-arc of the form (x, v) is found, we have found a cycle of length equal to (1 + the depth of x in the search tree).
3. Stop and Return the length found.

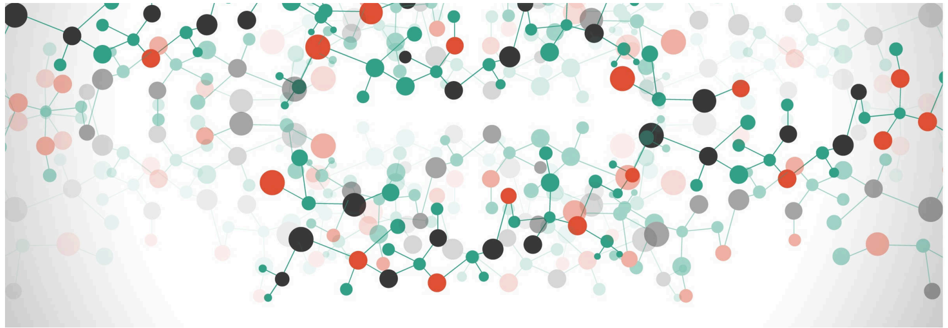
- Run the above procedure on every node and pick the length of the shortest cycle

SUMMARY

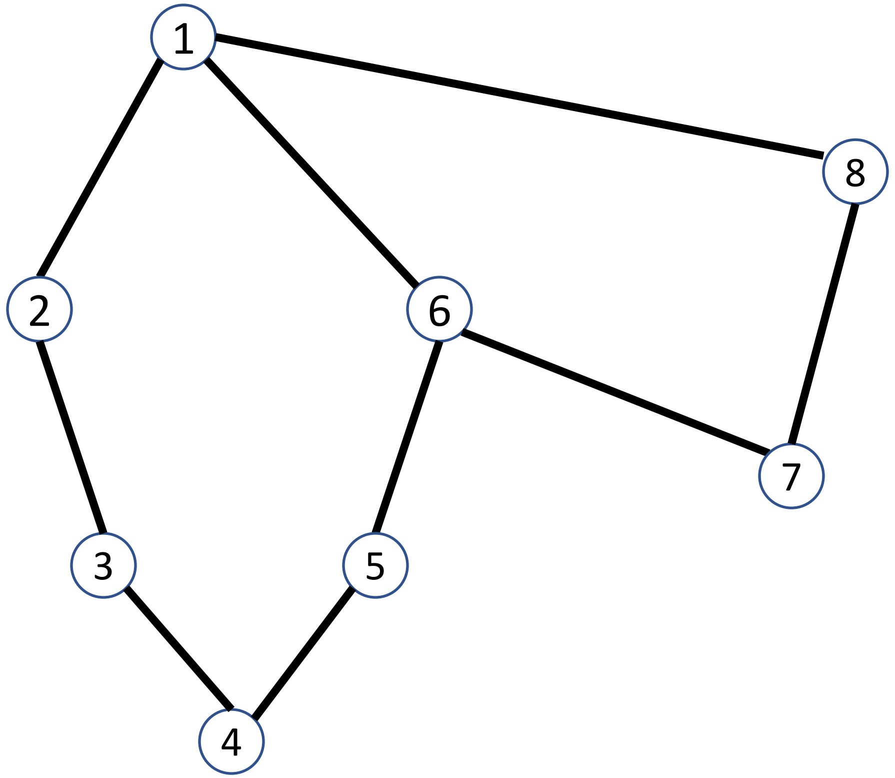
- Terminology
  - Cycle
  - Girth

- DFS Application: Cycle Detection
- BFS Application: Girth Detection

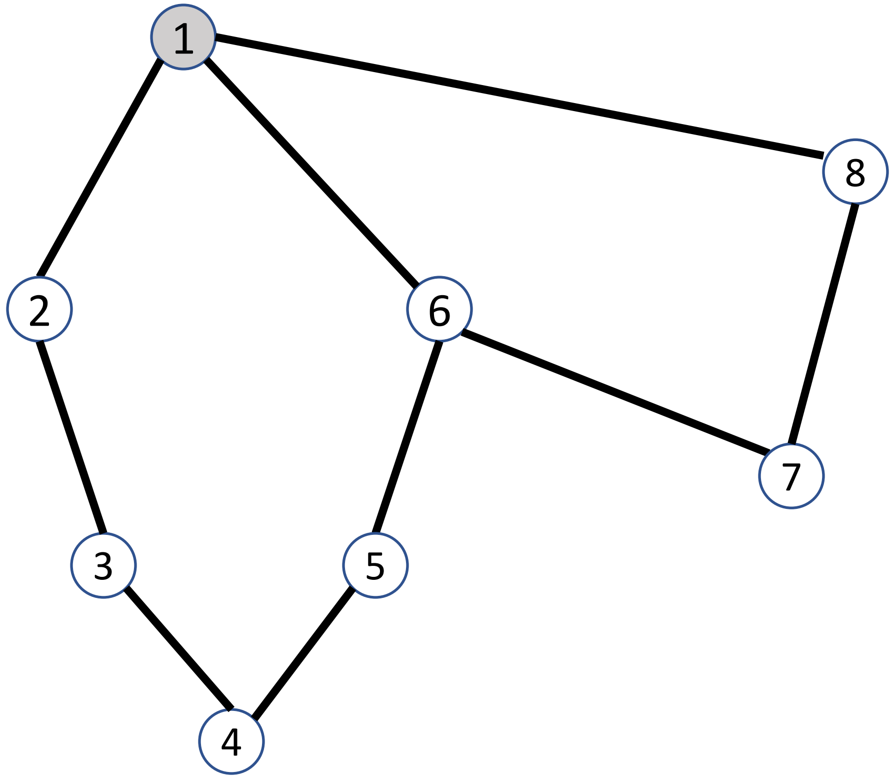
- Correctness of Girth Computation
- Illustrative Examples



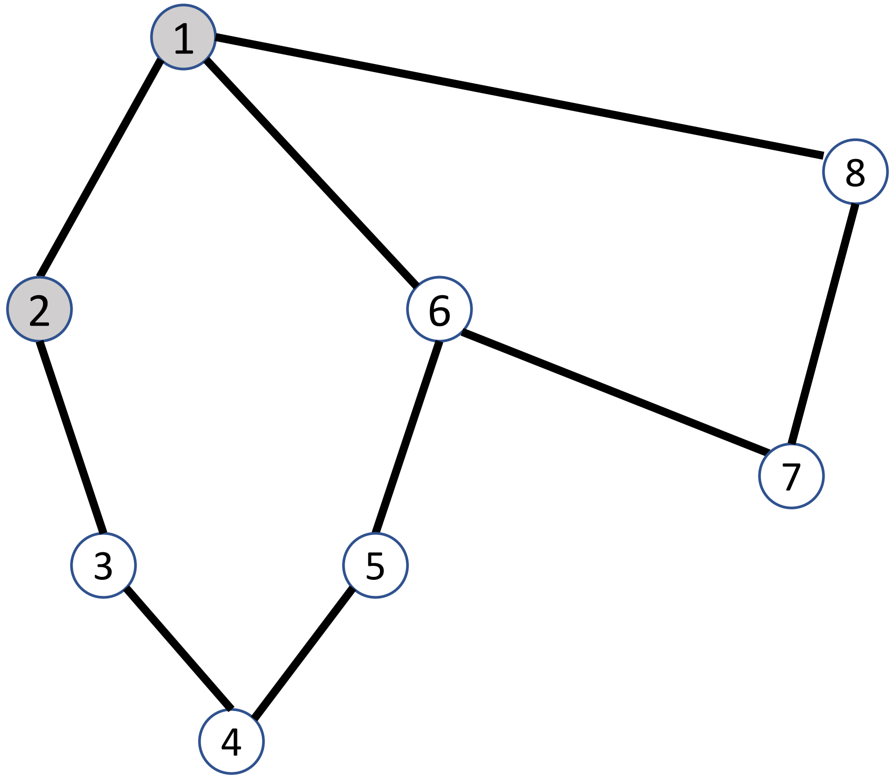
Exercise: Finding Cycle involving Vertex 1 using DFS



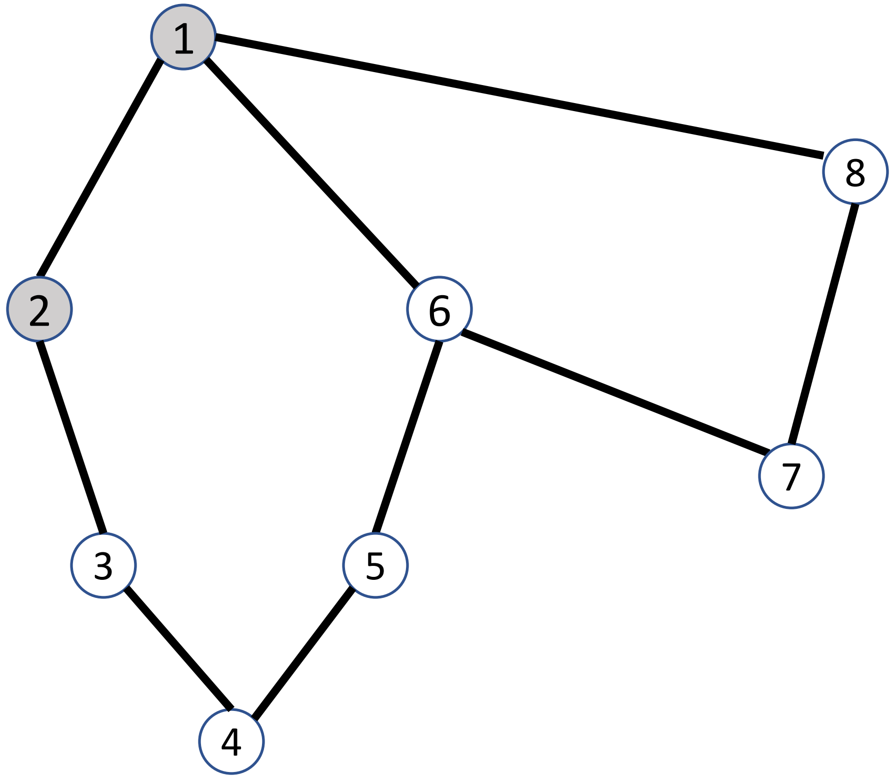
Exercise: Finding Cycle involving Vertex 1 using DFS



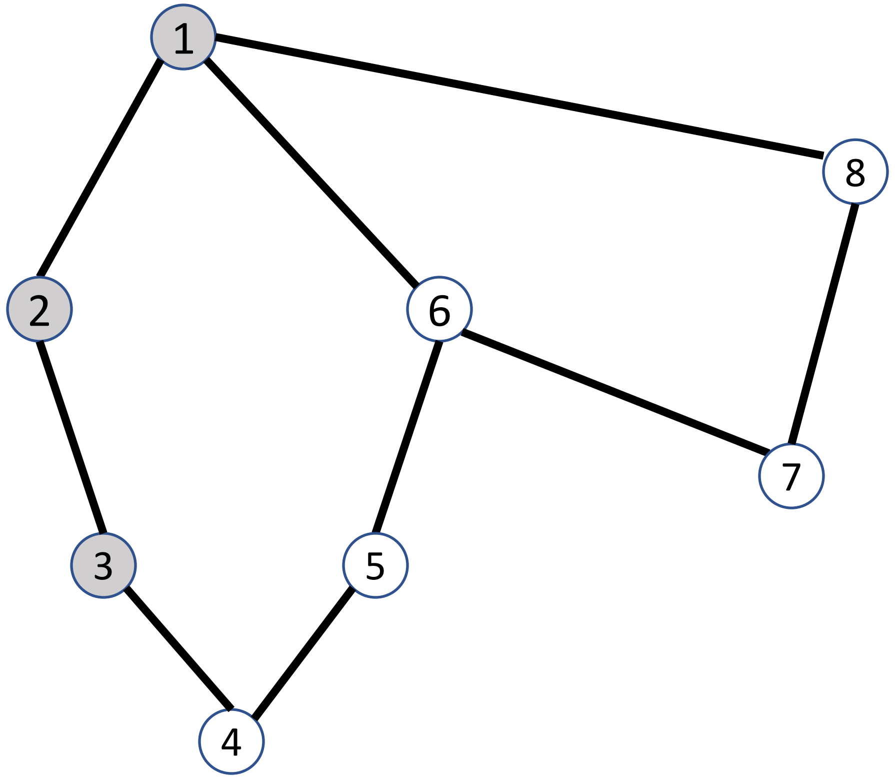
Exercise: Finding Cycle involving Vertex 1 using DFS



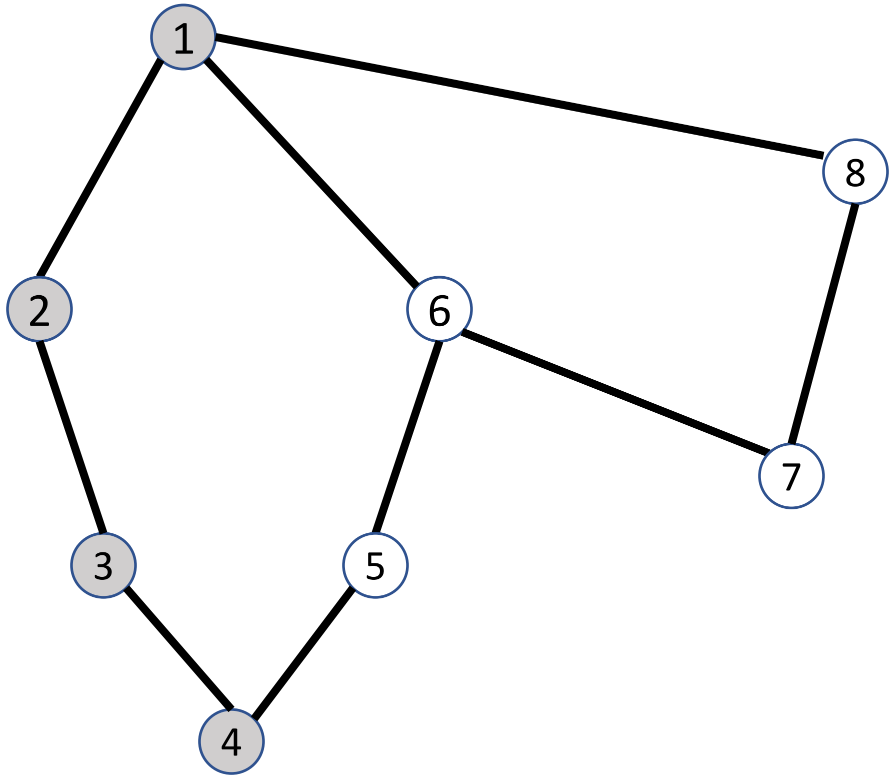
Exercise: Finding Cycle involving Vertex 1 using DFS



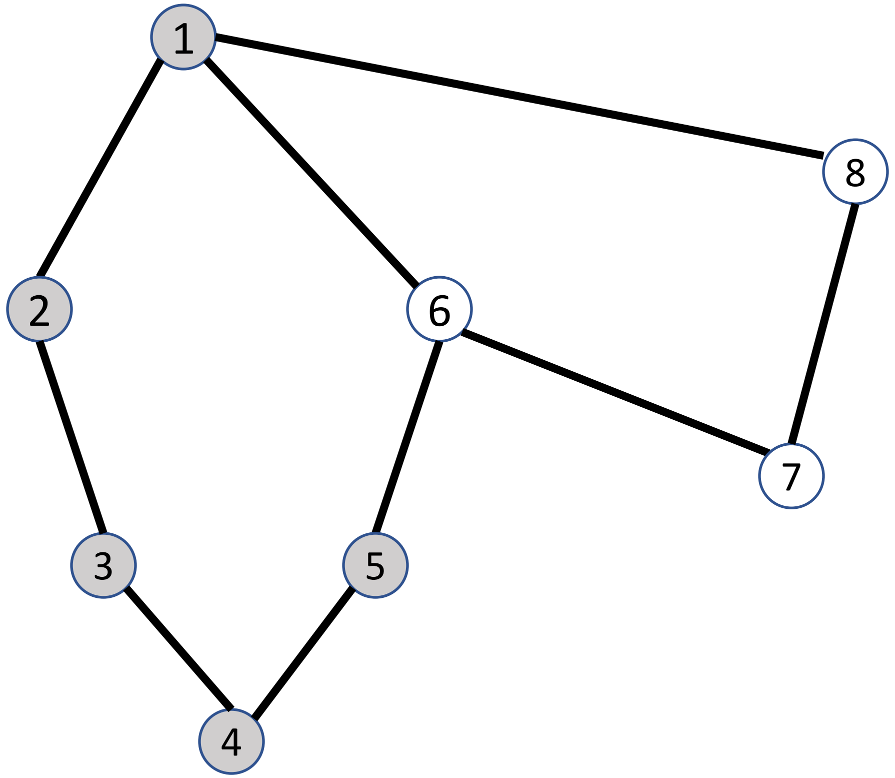
Exercise: Finding Cycle involving Vertex 1 using DFS



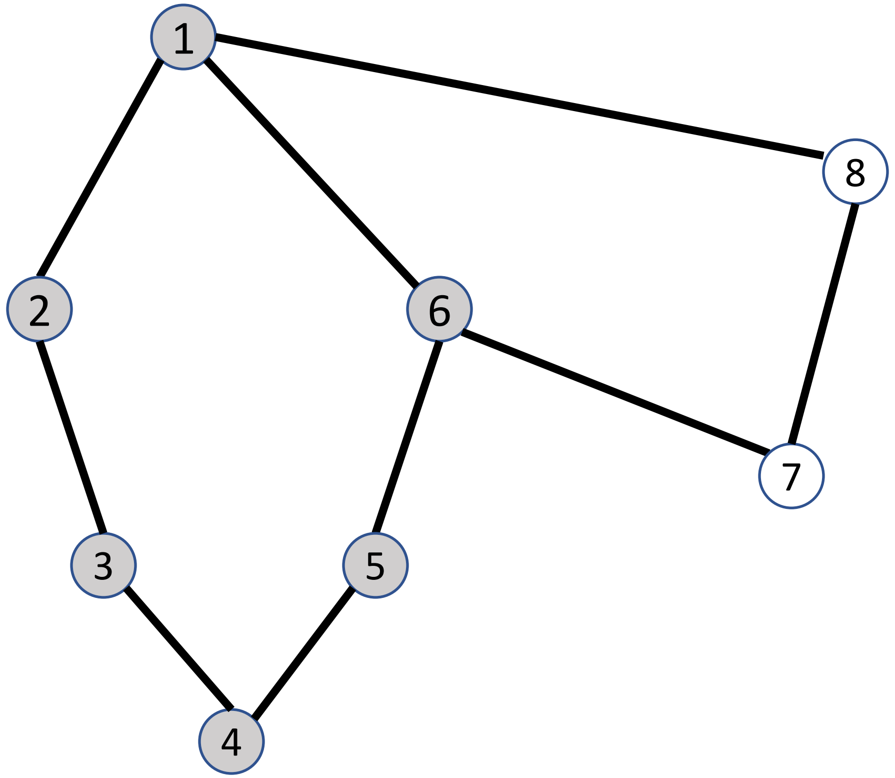
Exercise: Finding Cycle involving Vertex 1 using DFS



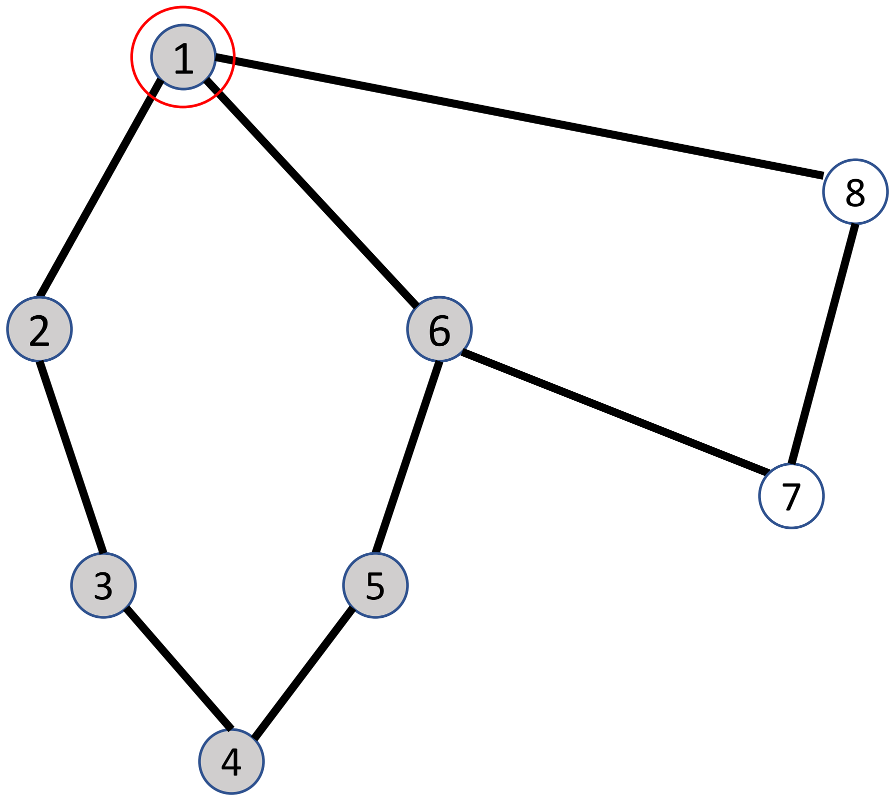
Exercise: Finding Cycle involving Vertex 1 using DFS



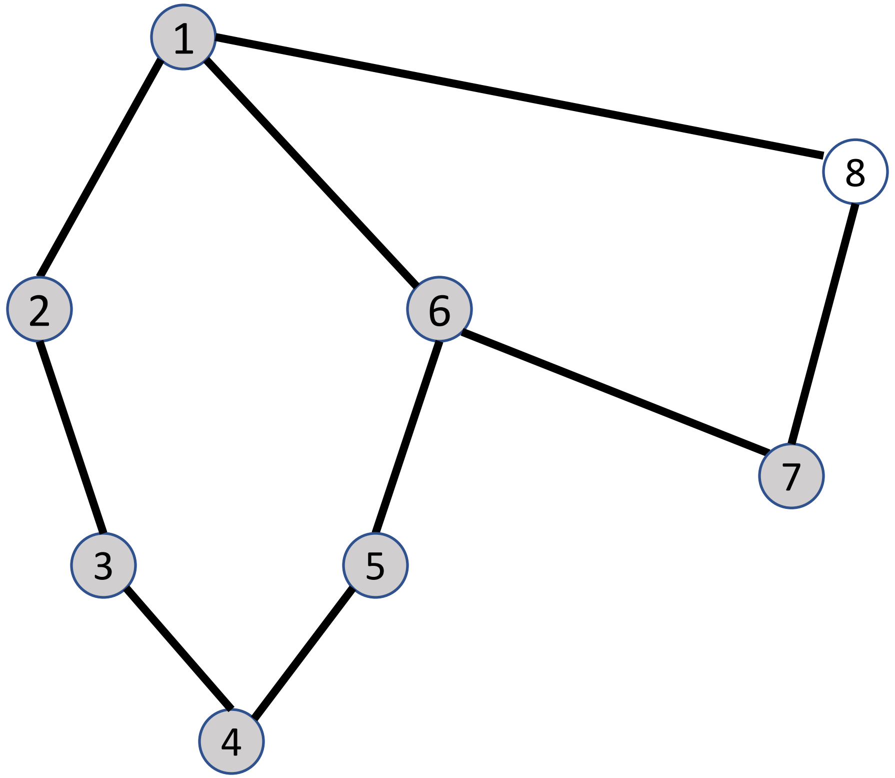
Exercise: Finding Cycle involving Vertex 1 using DFS



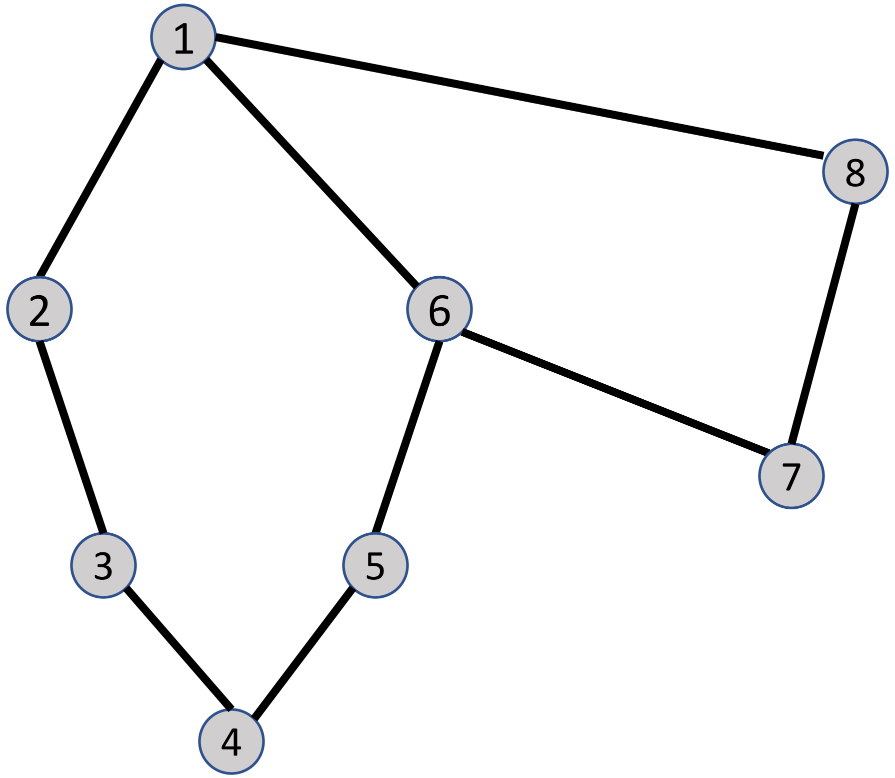
Exercise: Finding Cycle involving Vertex 1 using DFS



Exercise: Finding Cycle involving Vertex 1 using DFS



Exercise: Finding Cycle involving Vertex 1 using DFS



Exercise: Finding Cycle involving Vertex 1 using DFS