Fundamentals of Artificial Intelligence

Laboratory

Dr. Mauro Dragoni

Department of Information Engineering and Computer Science

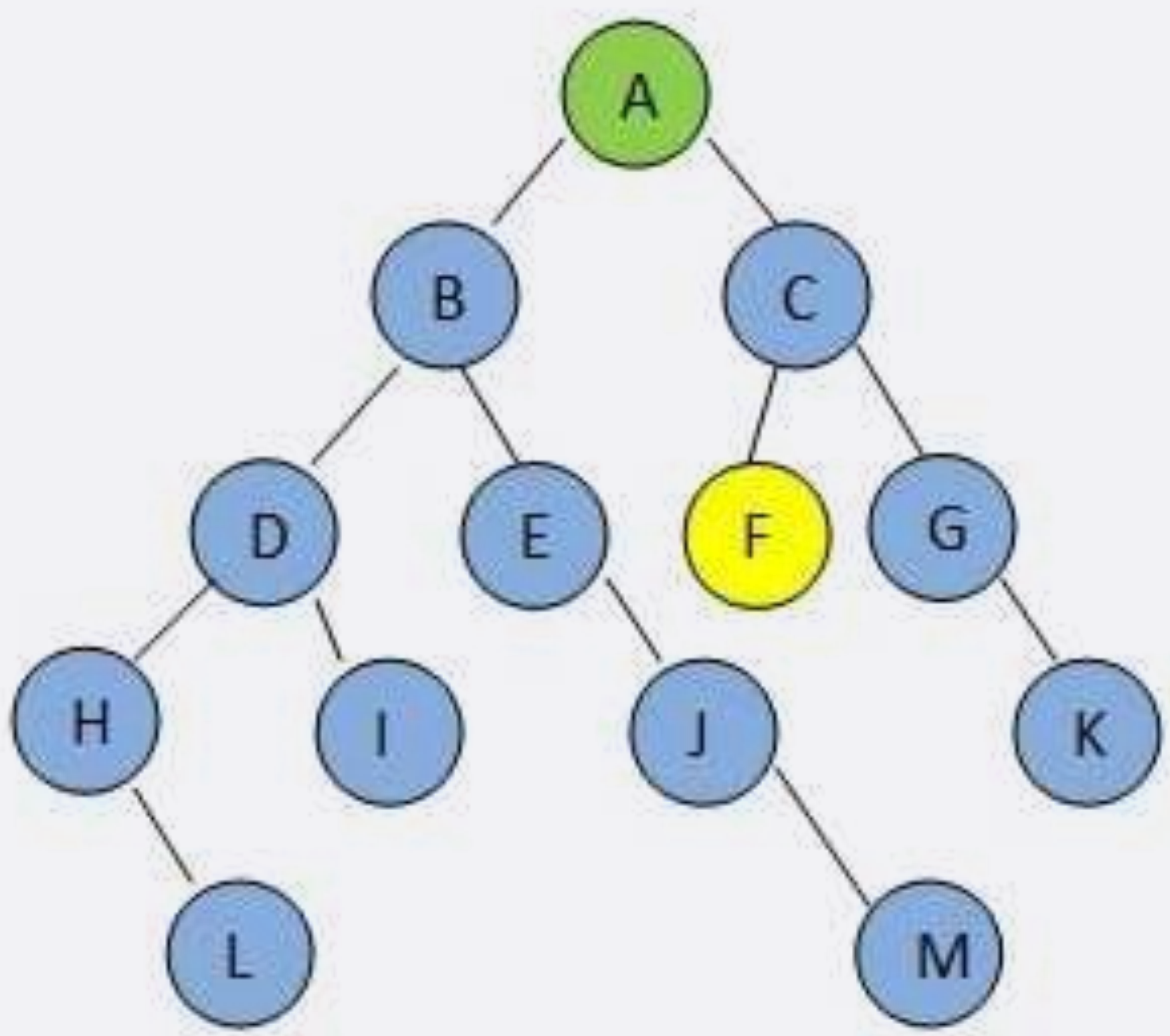
Academic Year 2020/2021

Some points…

- Tree search vs. Graph search.
- Goal test applied when a node is generated vs. when a node is explored.
- Partially expansion or total expansion (DFS only).

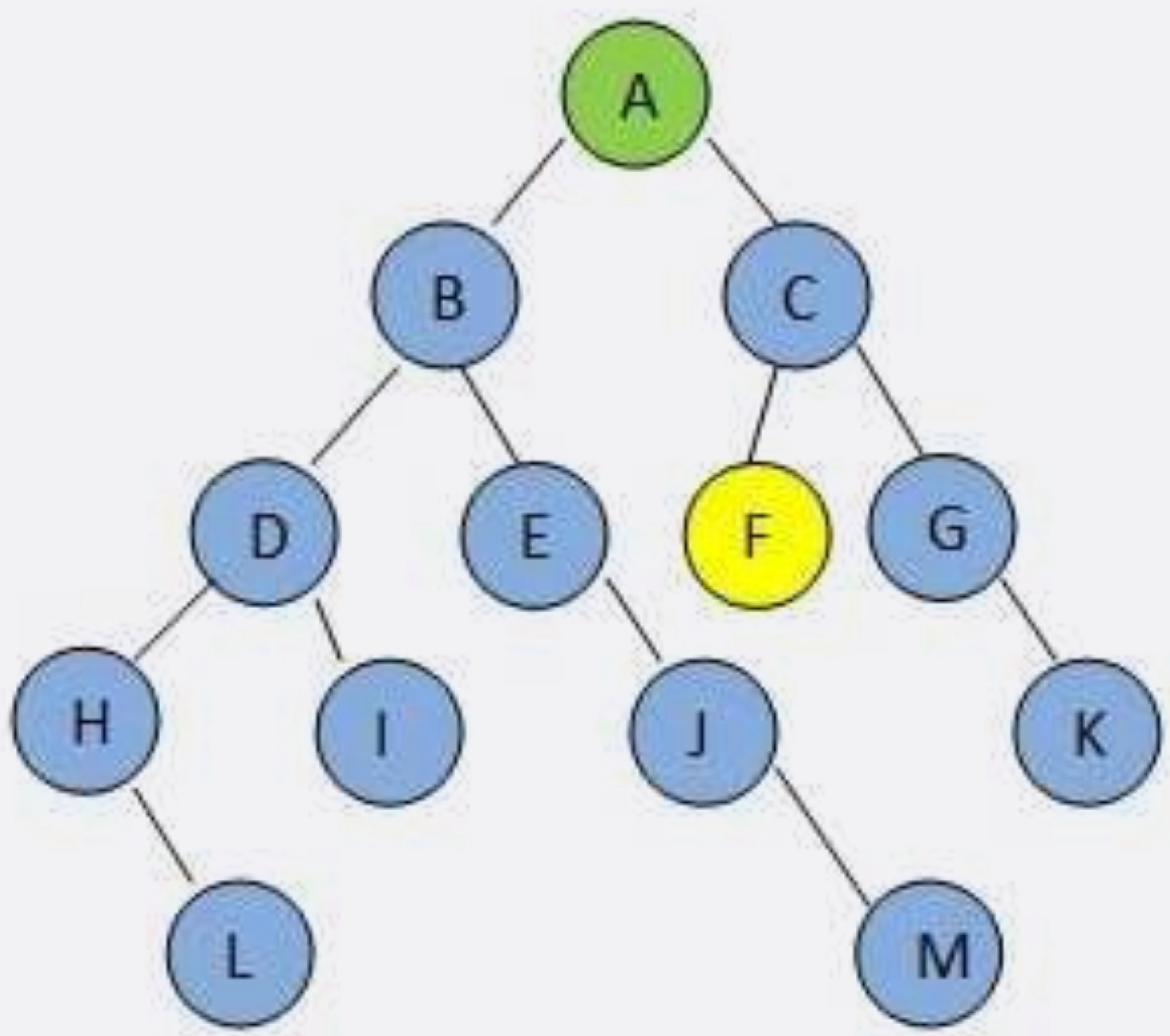
Exercise 3.1

- In the following graphs, assume that if there is ever a choice amongst multiple nodes, both the BFS and DFS algorithms will choose the left-most node first.
- Starting from the green node at the top, which algorithm will visit the least number of nodes before visiting the yellow goal node?



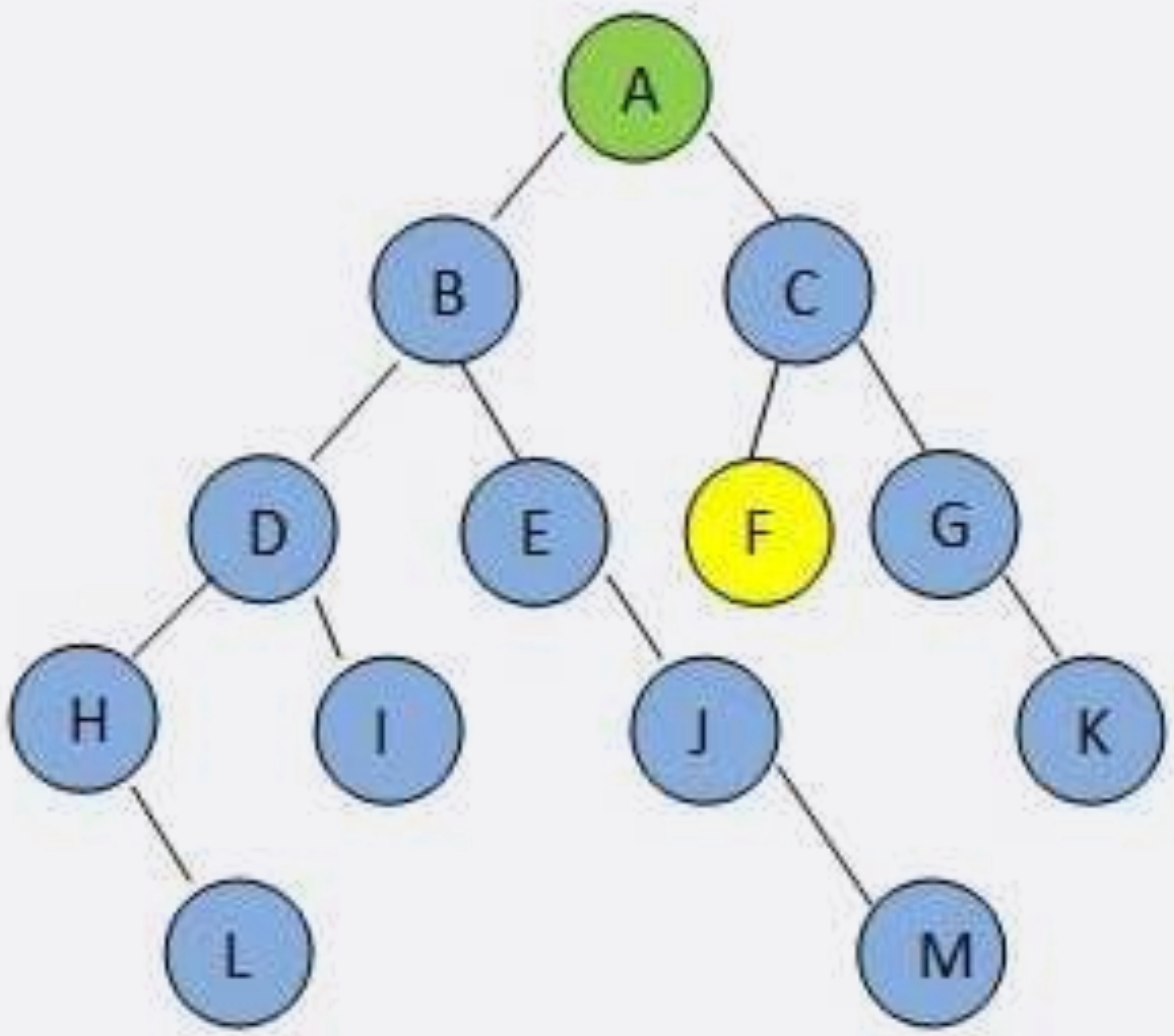
Exercise 3.1

A. BFS
B. DFS
C. Neither BFS nor DFS will ever encounter the goal node in this graph.
D. BFS and DFS encounter same number of nodes before encounter the goal node



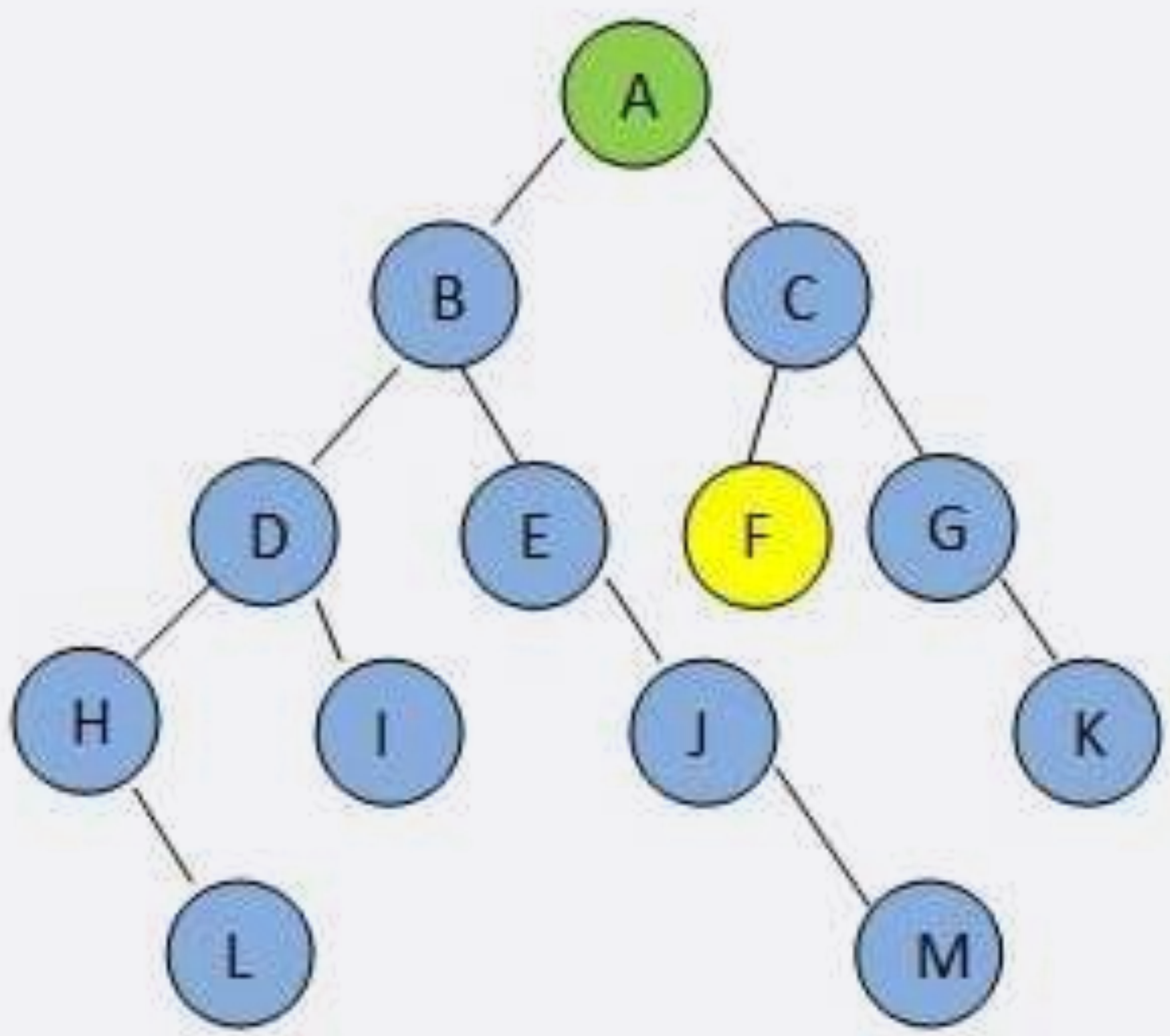
Exercise 3.1

A. **BFS**
B. DFS
C. Neither BFS nor DFS will ever encounter the goal node in this graph.
D. BFS and DFS encounter same number of nodes before encounter the goal node



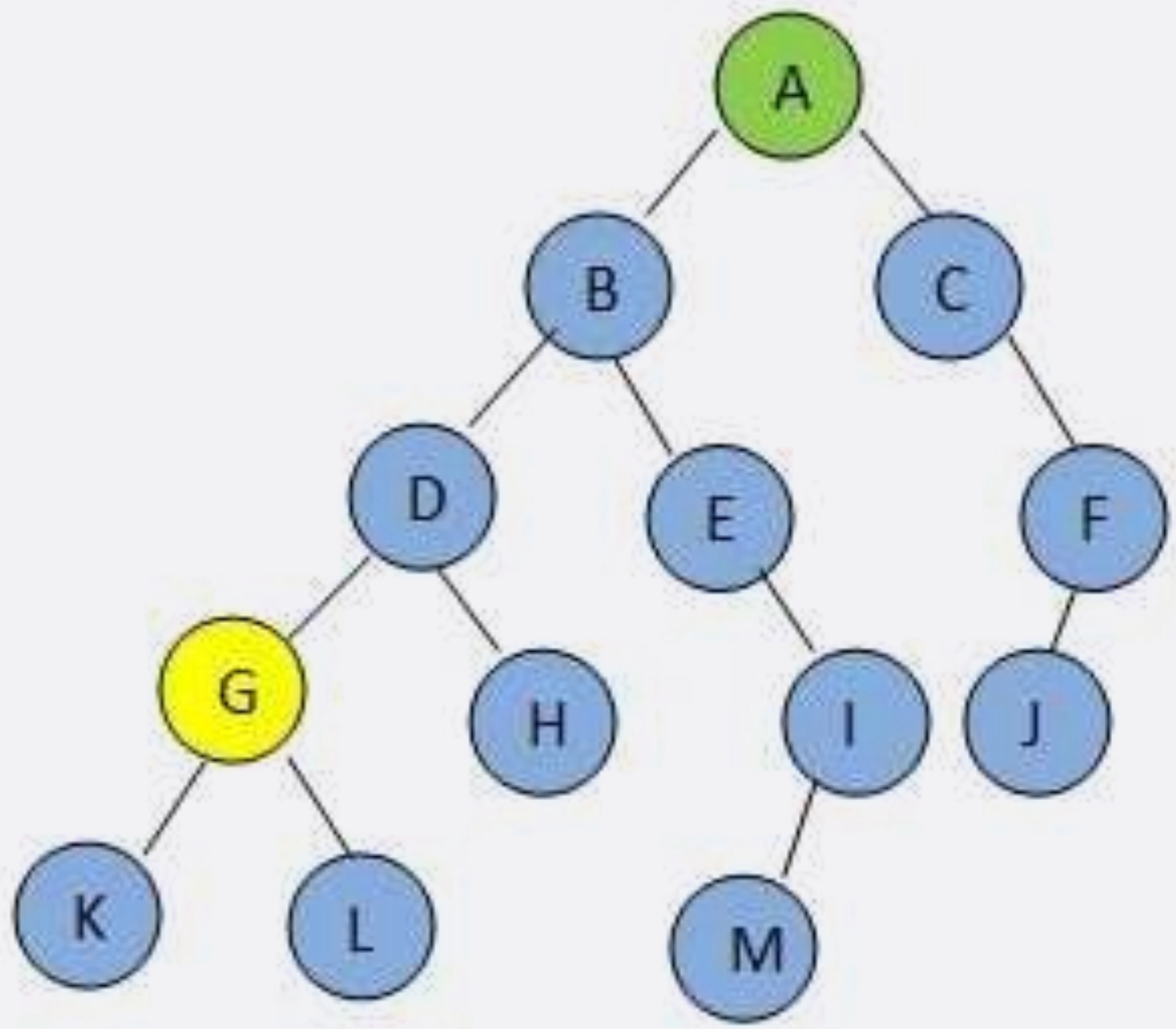
Exercise 3.1

- For BFS algorithm, visiting a node's siblings before its children, while in DFS algorithm, visiting a node's children before its siblings. Before countering goal node F:
- BFS algorithm encounters nodes: ABCDE
- DFS algorithm encounters nodes: ABDHLIEJMC



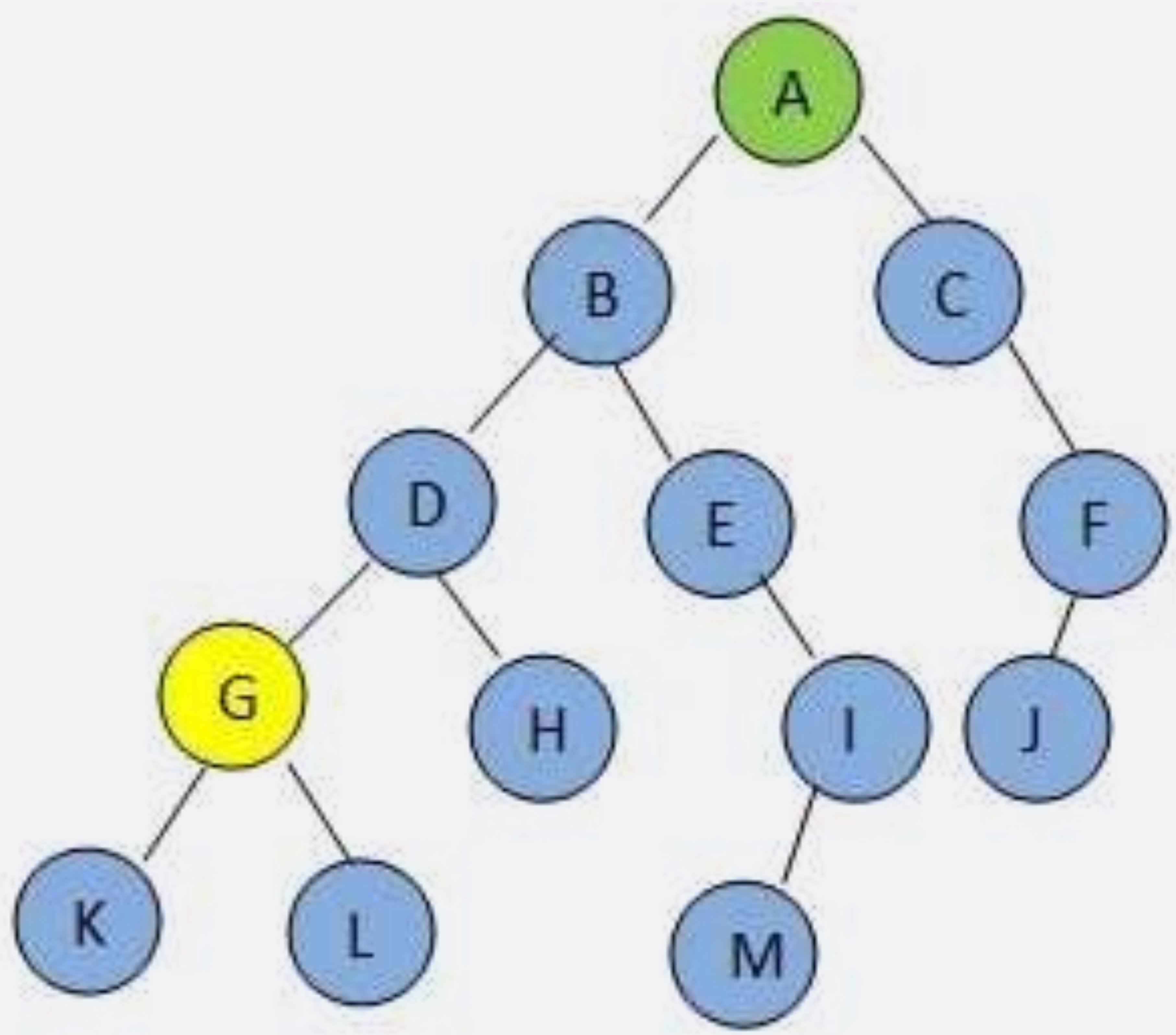
Exercise 3.2

- In the following graphs, assume that if there is ever a choice amongst multiple nodes, both the BFS and DFS algorithms will choose the left-most node first.
- Starting from the green node at the top, which algorithm will visit the least number of nodes before visiting the yellow goal node?



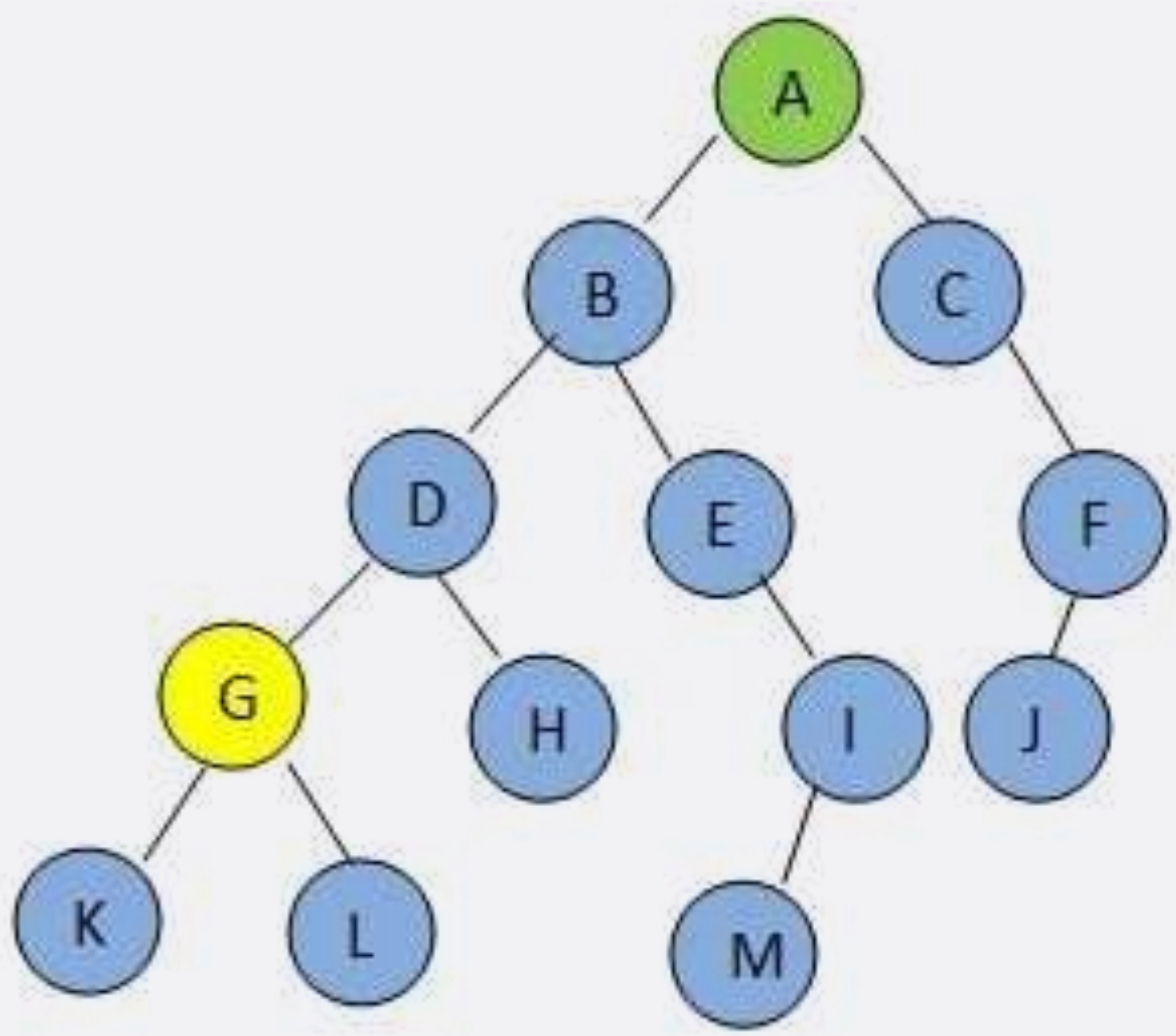
Exercise 3.2

A. BFS
B. DFS
C. Neither BFS nor DFS will ever encounter the goal node in this graph.
D. BFS and DFS encounter same number of nodes before encounter the goal node



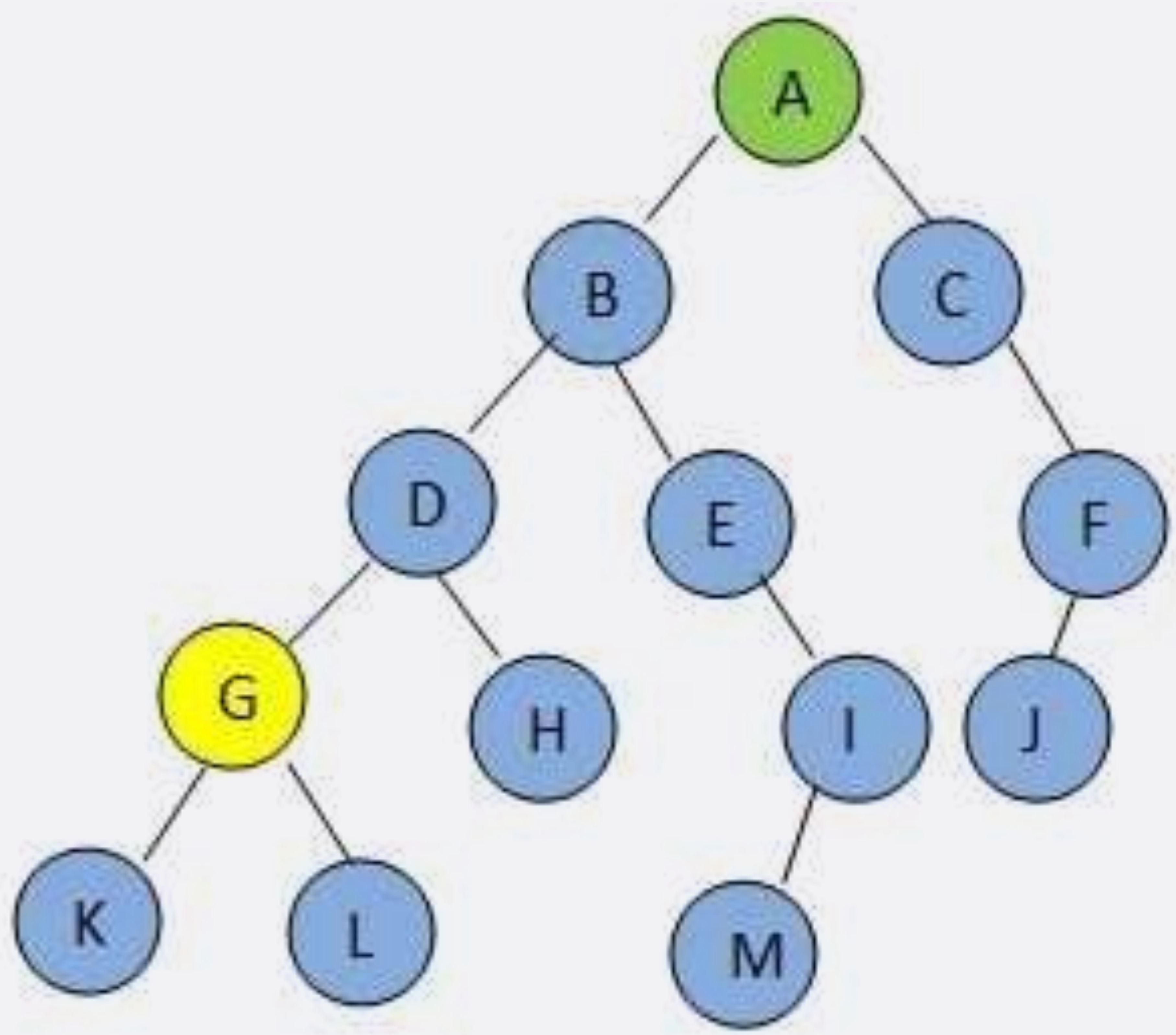
Exercise 3.2

A. BFS
B. DFS
C. Neither BFS nor DFS will ever encounter the goal node in this graph.
D. BFS and DFS encounter same number of nodes before encounter the goal node



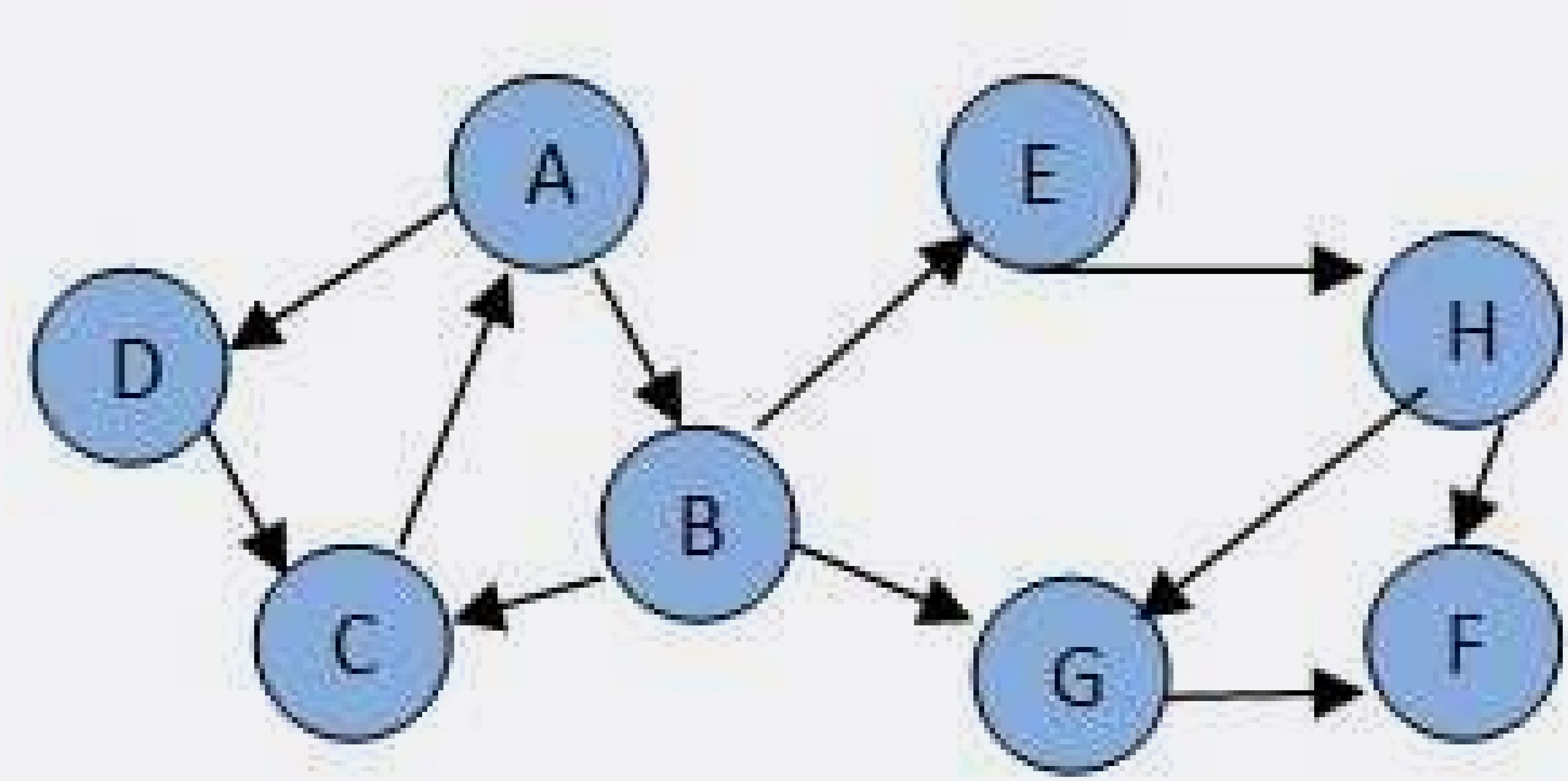
Exercise 3.2

- For BFS algorithm, visiting a node’s siblings before its children, while in DFS algorithm, visiting a node’s children before its siblings. Before countering goal node G:
- BFS algorithm encounters nodes: ABCDEF
- DFS algorithm encounters nodes: ABD



Exercise 3.3

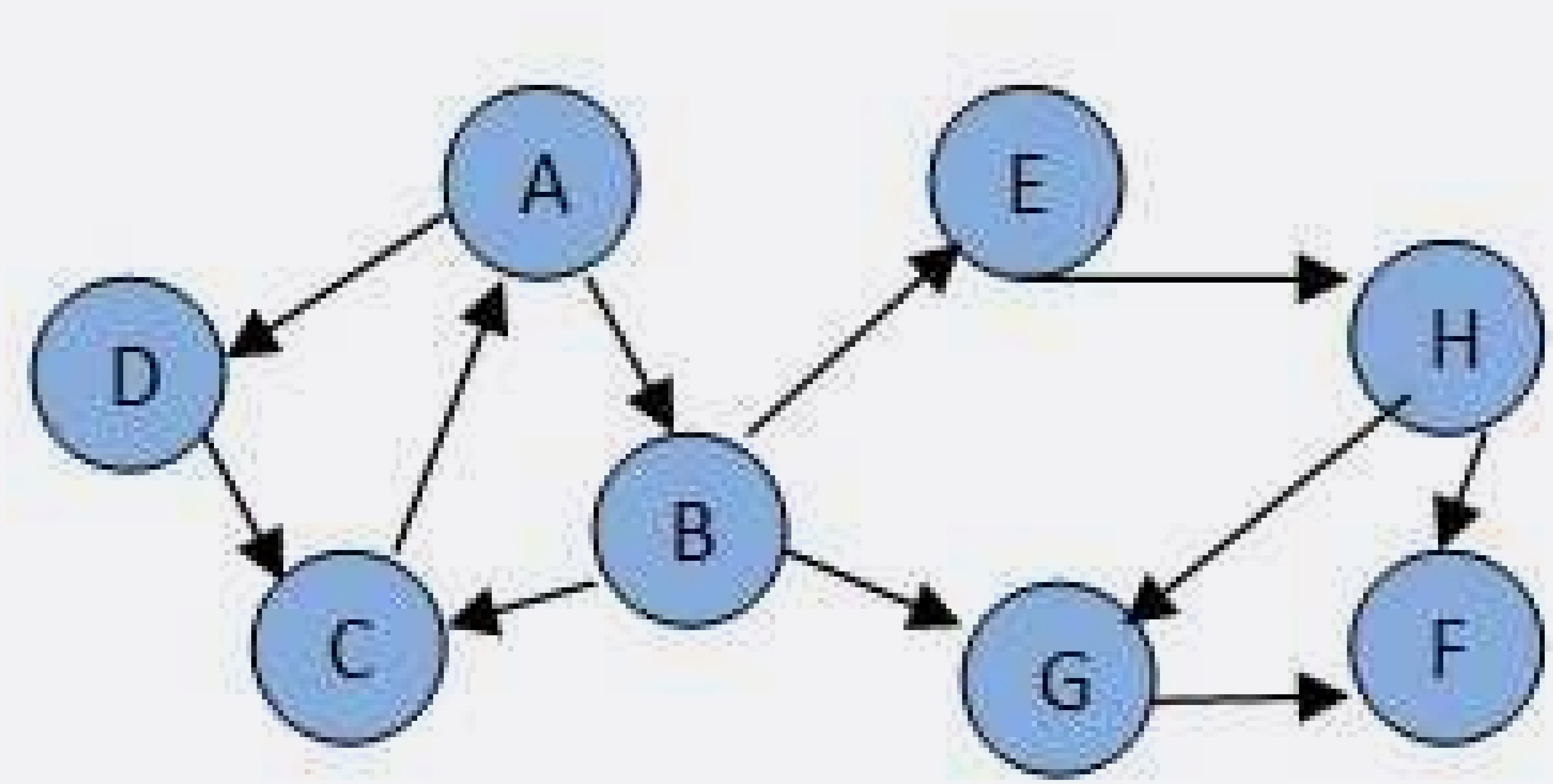
- Consider the following graph. If there is ever a decision between multiple neighbor nodes in the BFS or DFS algorithms, assume we always choose the letter closest to the beginning of the alphabet first. In what order will the nodes be visited using a Breadth First Search? In what order will the nodes be visited using a Depth First Search?



Exercise 3.3

- In what order will the nodes be visited using a Breadth First Search?
  The answer is: **ABDCEGHF**

- In what order will the nodes be visited using a Depth First Search?
  The answer is: **ABCEHFGD**

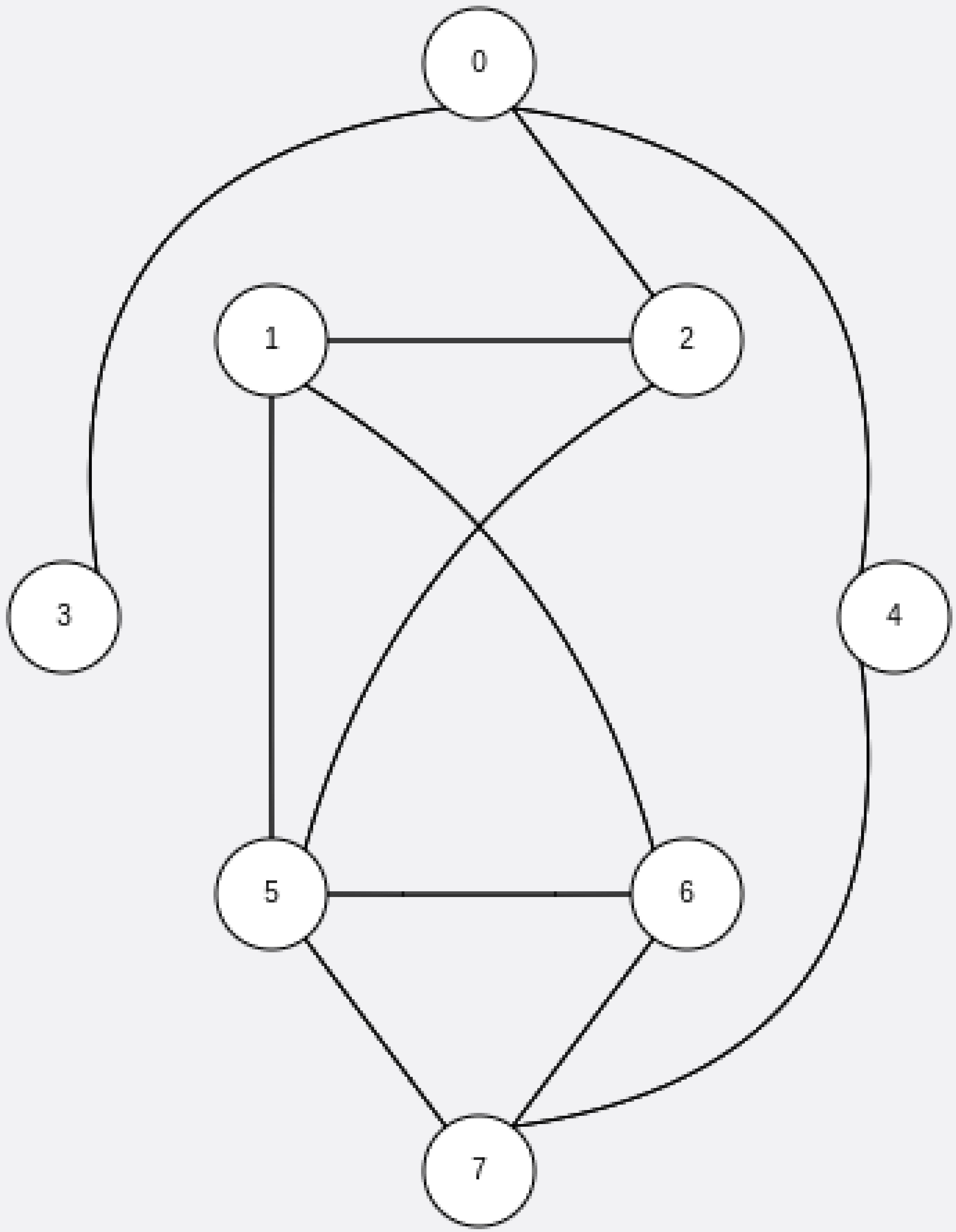


How to practice?

GO HERE:

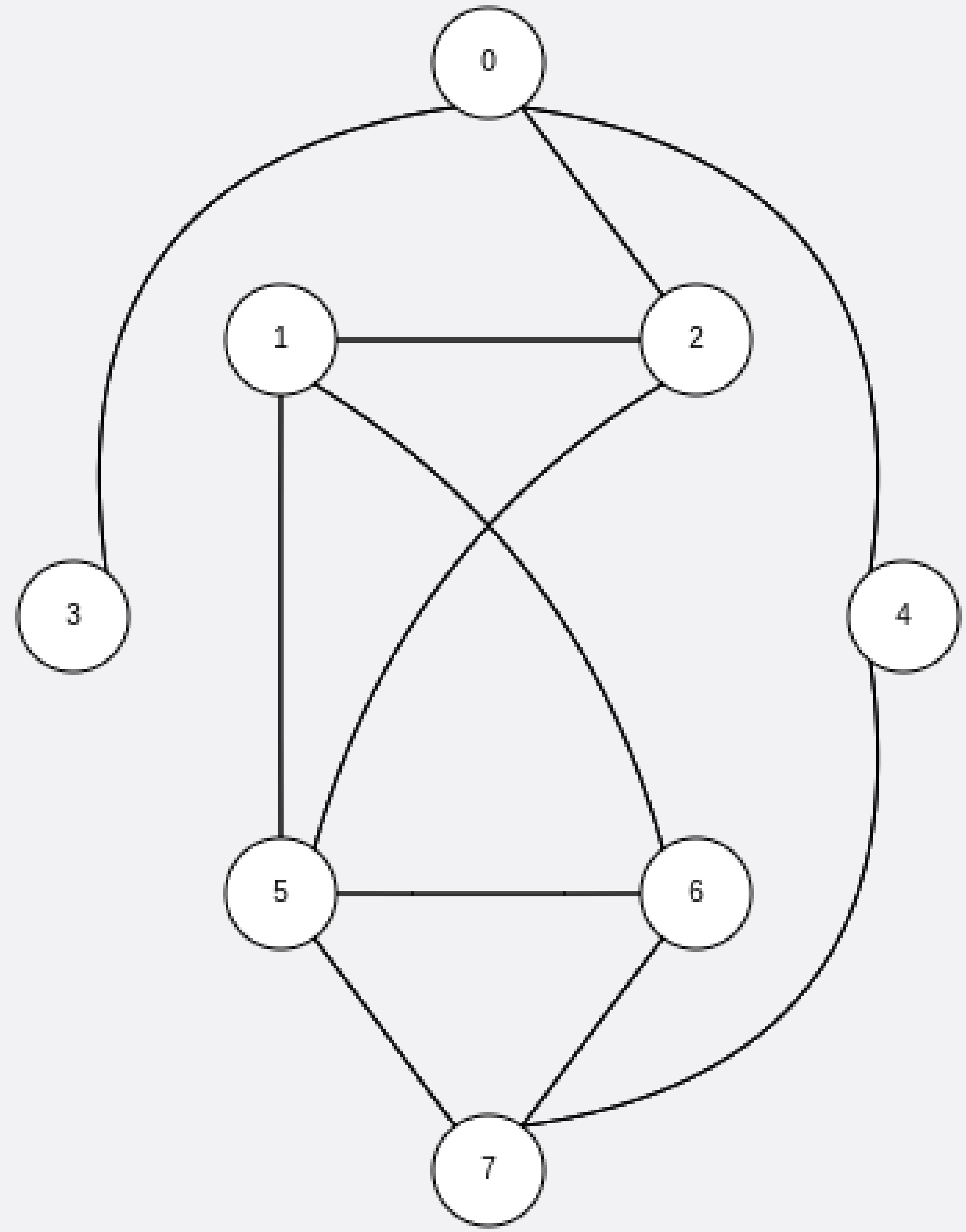
https://www.cs.usfca.edu/~galles/visualization/BFS.html

Exercise 3.4



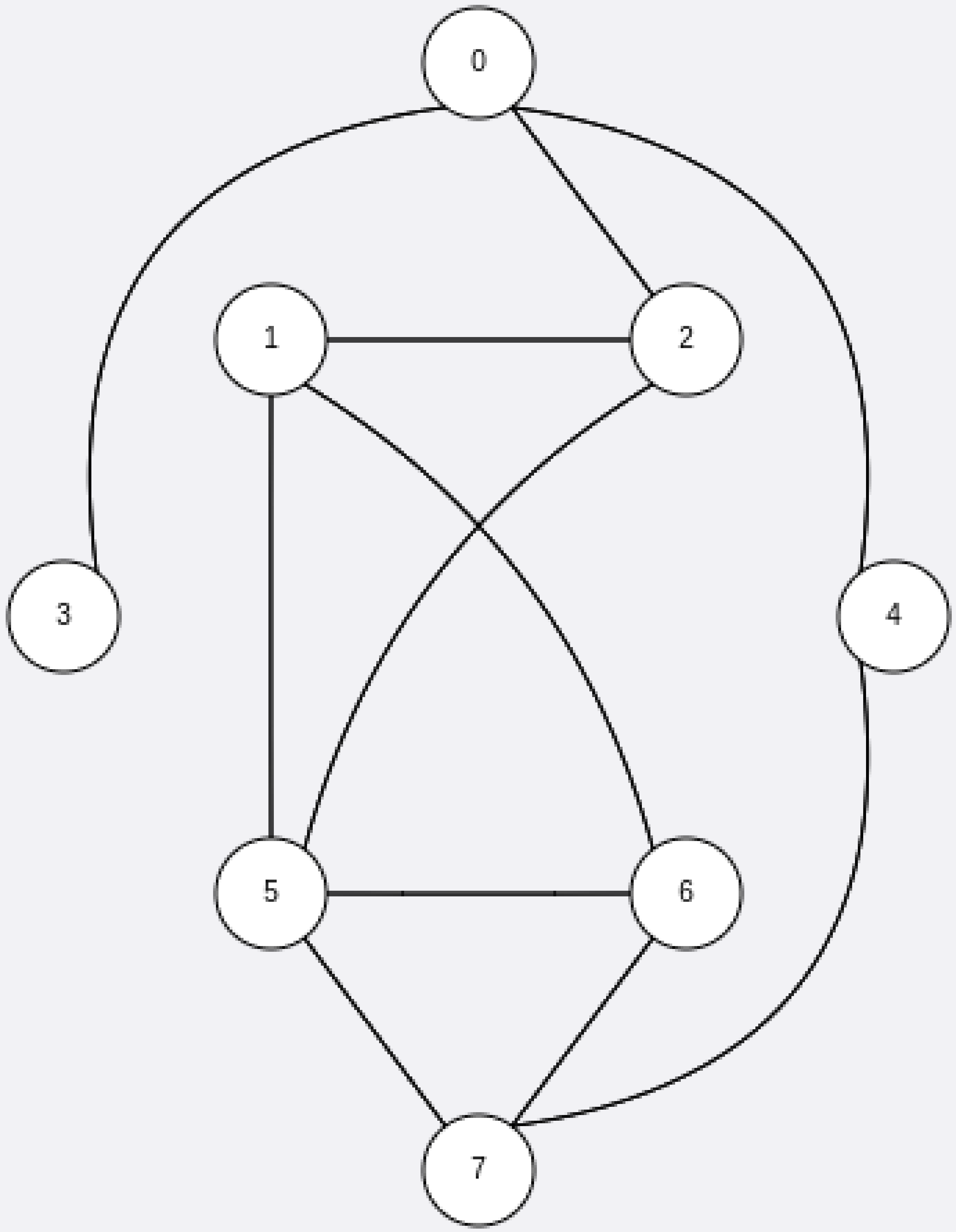
Exercise 3.4 – Solution - BFS

Step	Queue	Exp. Node	Gen. Nodes
0	0	0	2, 3, 4
1	2, 3, 4	2	**0**, 1, 5
2	3, 4, 1, 5	3	-
3	4, 1, 5	4	**0**, 7
4	1, 5, 7	1	**5**, 6
5	5, 7, 6	5	**1**, **6**, **7**
6	7, 6	7	**4**, **5**, **6**
7	6	6	-



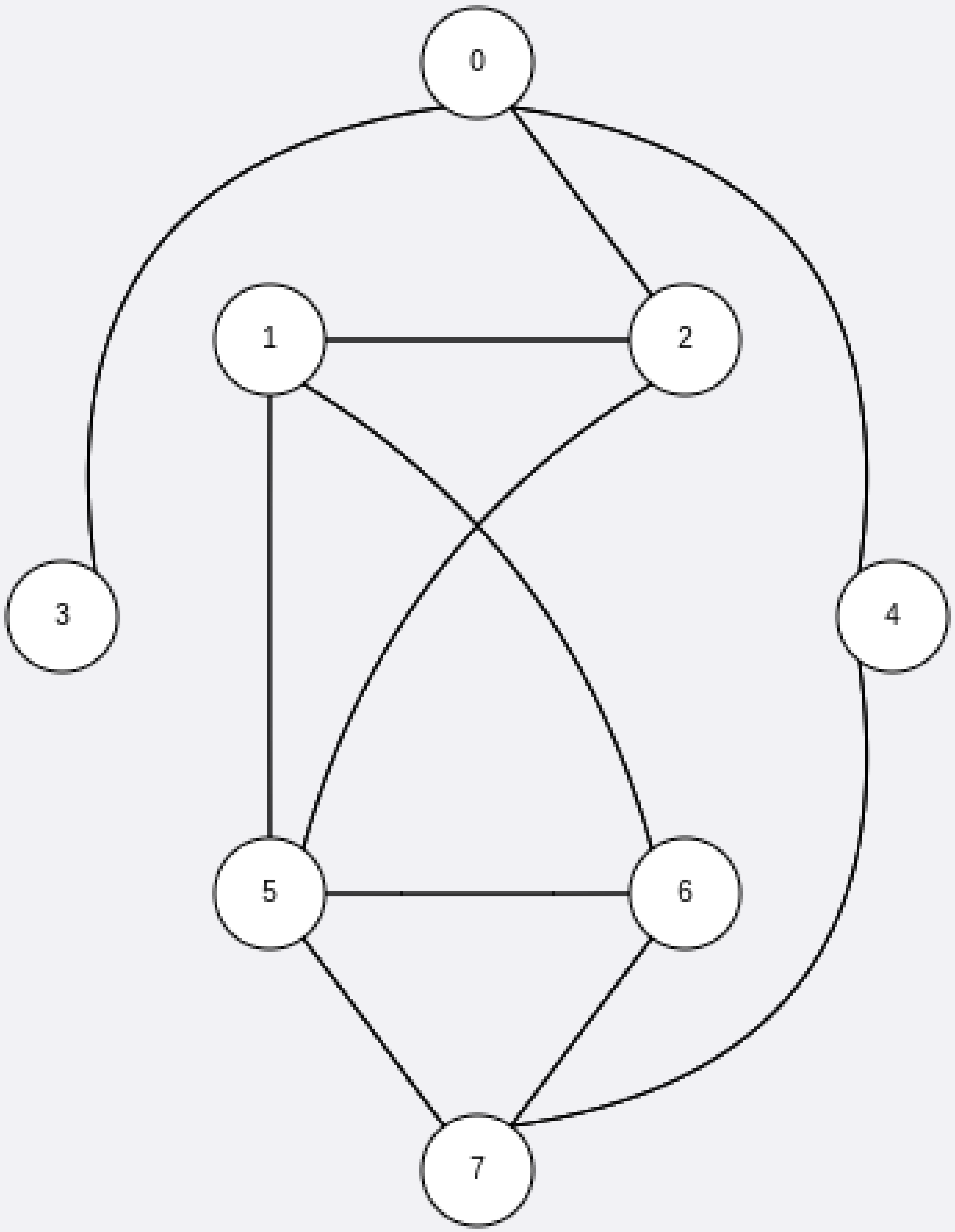
Exercise 3.4 – Solution - DFS

Step	Queue	Exp. Node	Gen. Nodes
0	0	0	2, 3, 4
1	2, 3, 4	2	**0**, 1, 5
2	1, 5, 3, 4	1	**2**, **5**, 6
3	5, 6, 3, 4	5	**1**, 6, 7
4	6, 7, 3, 4	6	**1**, **5**, 7
5	7, 3, 4	7	4, **5**, **6**
6	4, 3	4	**0**, **7**
7	3	3	-

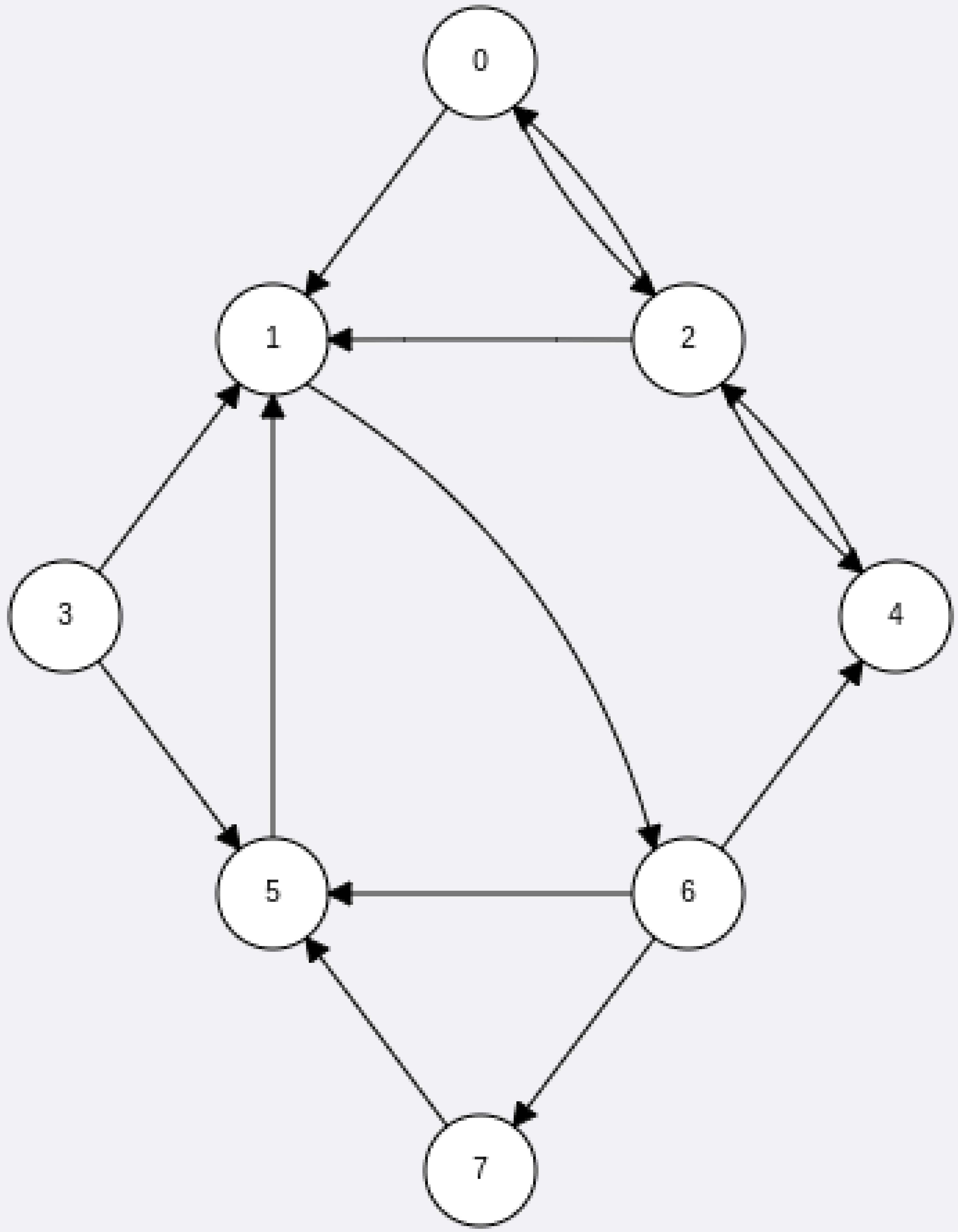


Exercise 3.4 - Solution

- **BFS:** 0 2 3 4 1 5 7 6
- **DFS:** 0 2 1 5 6 7 4 3

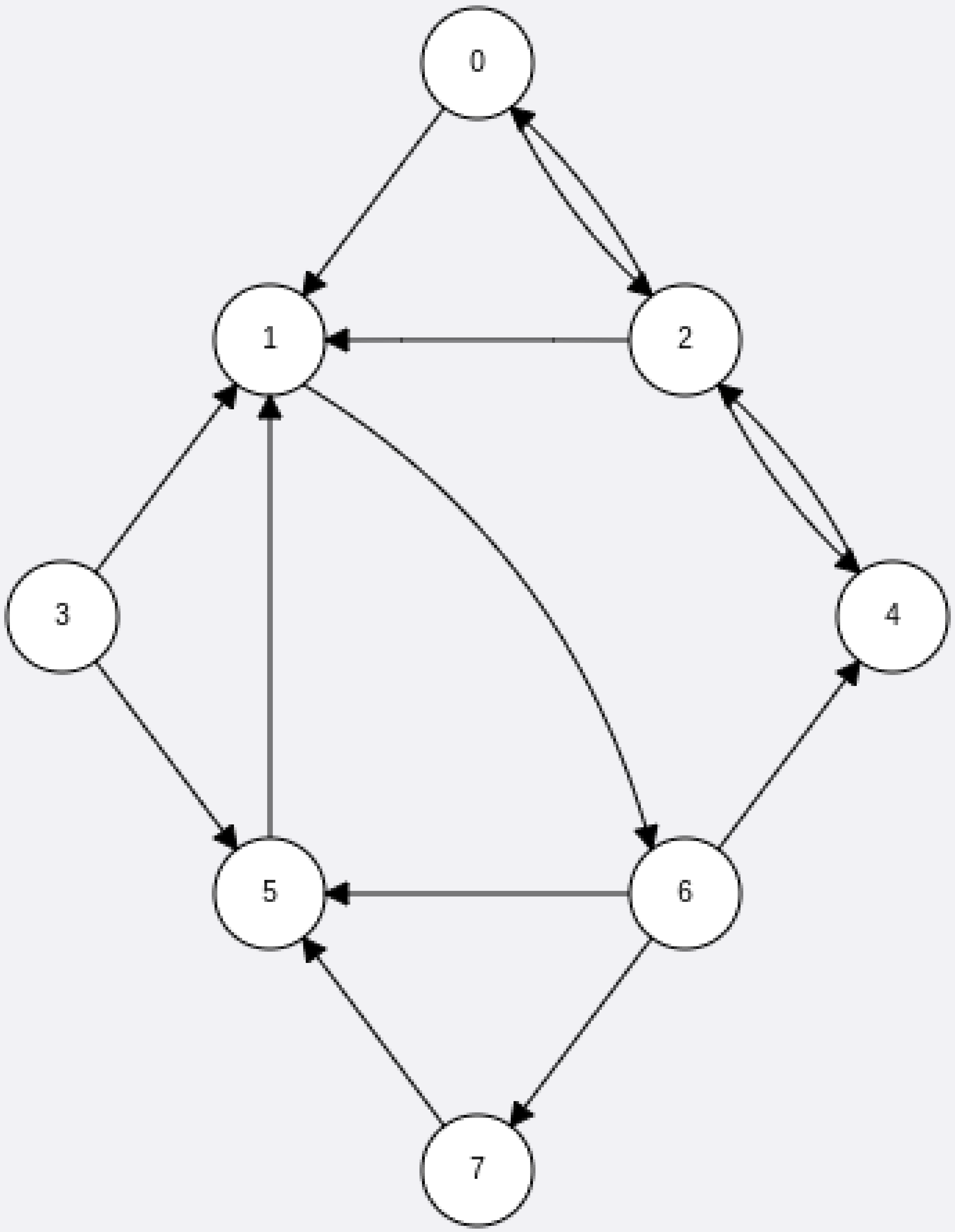


Exercise 3.5

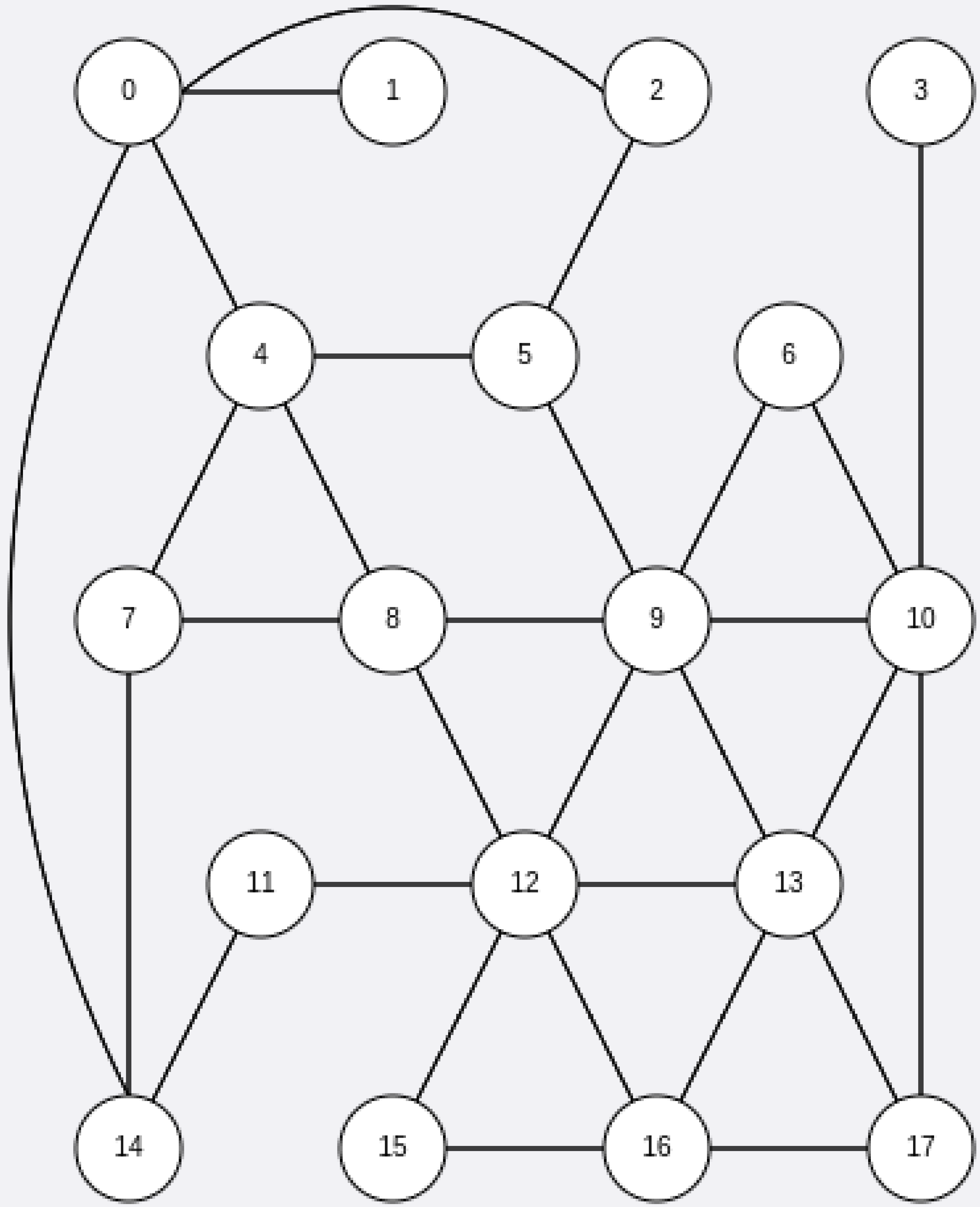


Exercise 3.5 - Solution

- **BFS:** 0 1 2 6 4 5 7
- **DFS:** 0 1 6 4 2 5 7

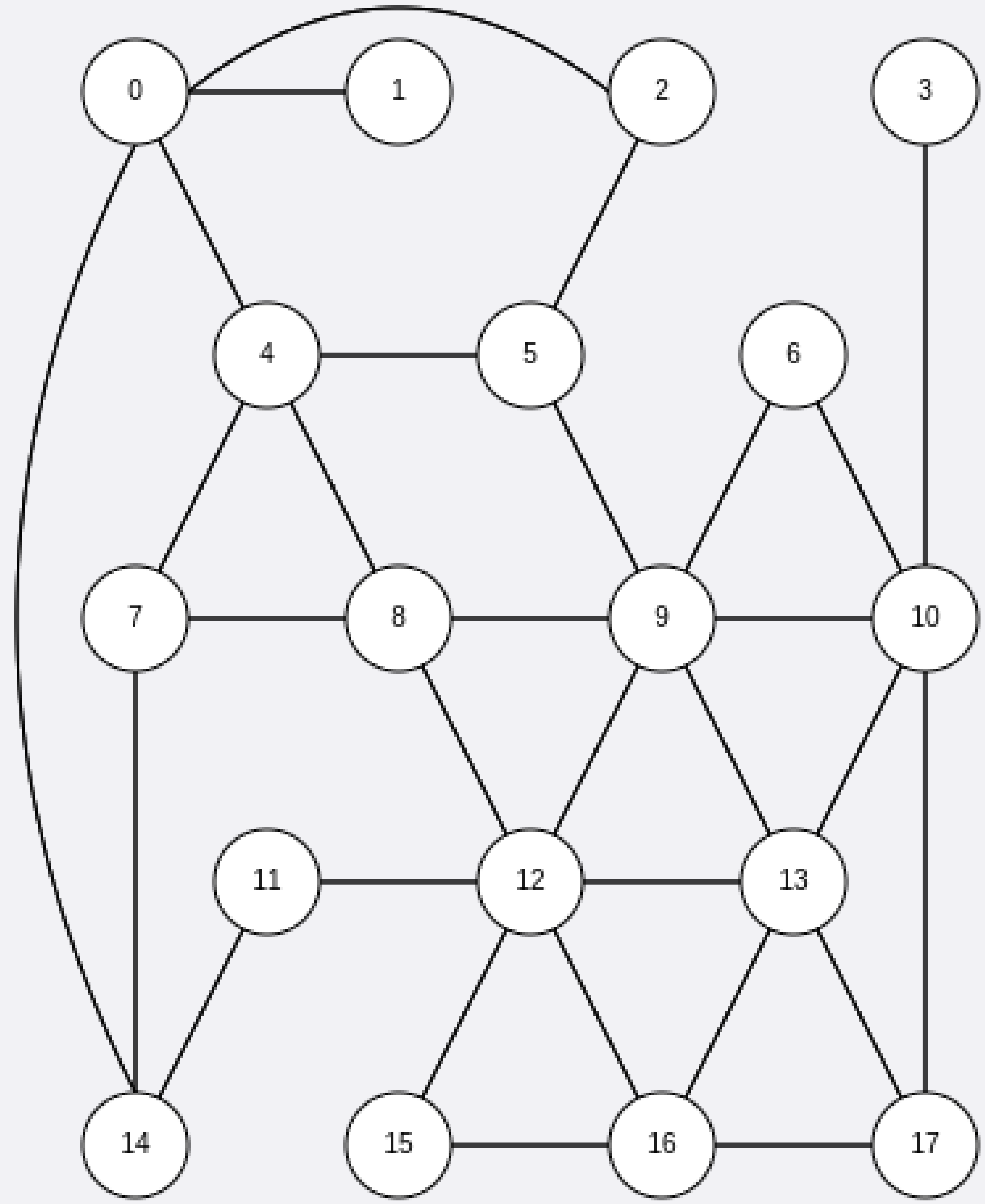


Exercise 3.6

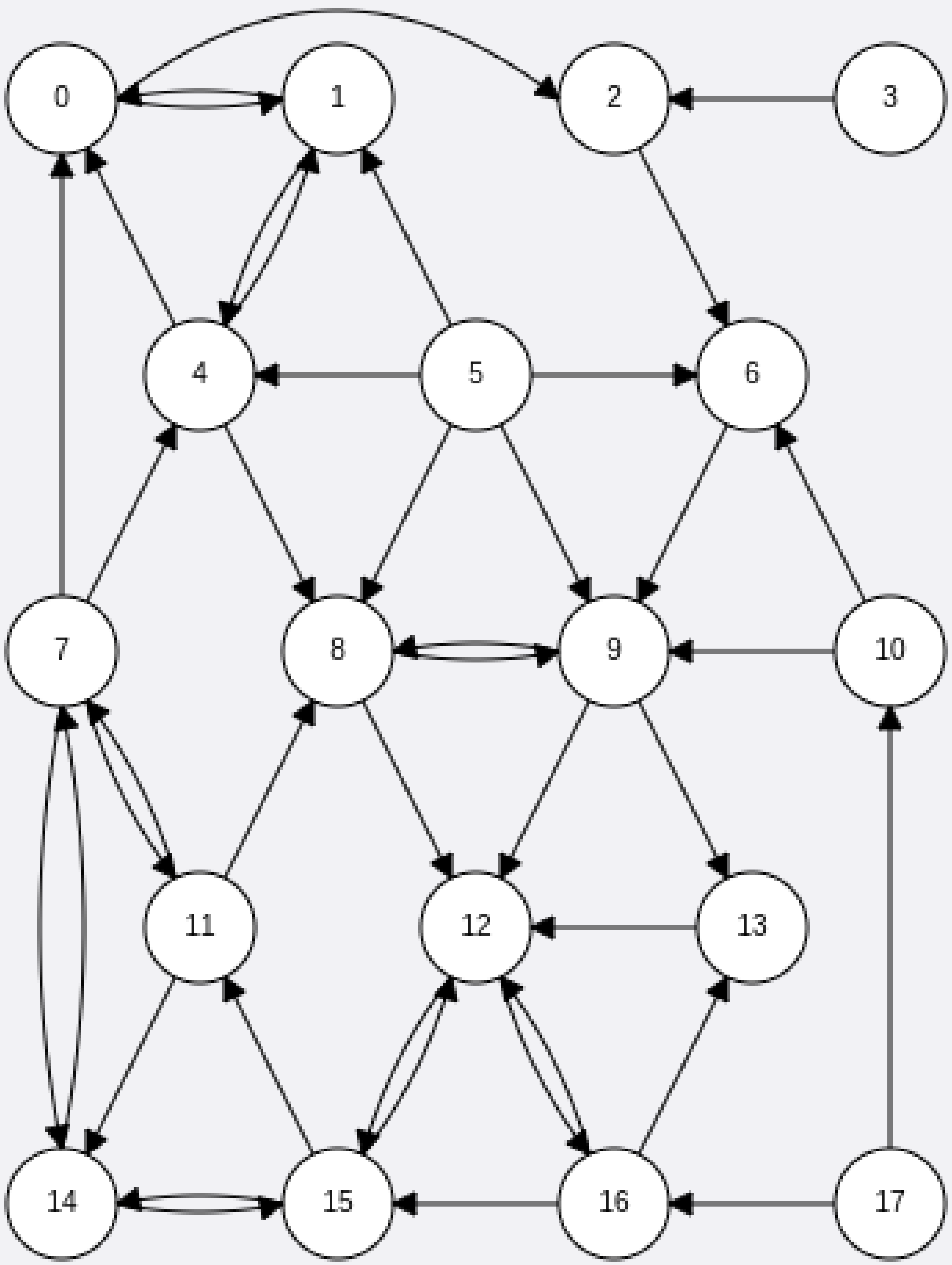


Exercise 3.6 - Solution

- **BFS:** 0 1 2 4 14 5 7 8 11 9
  12 6 10 13 15 16 3 17
- **DFS:** 0 1 2 5 4 7 8 9 6 10 3
  13 12 11 14 15 16 17

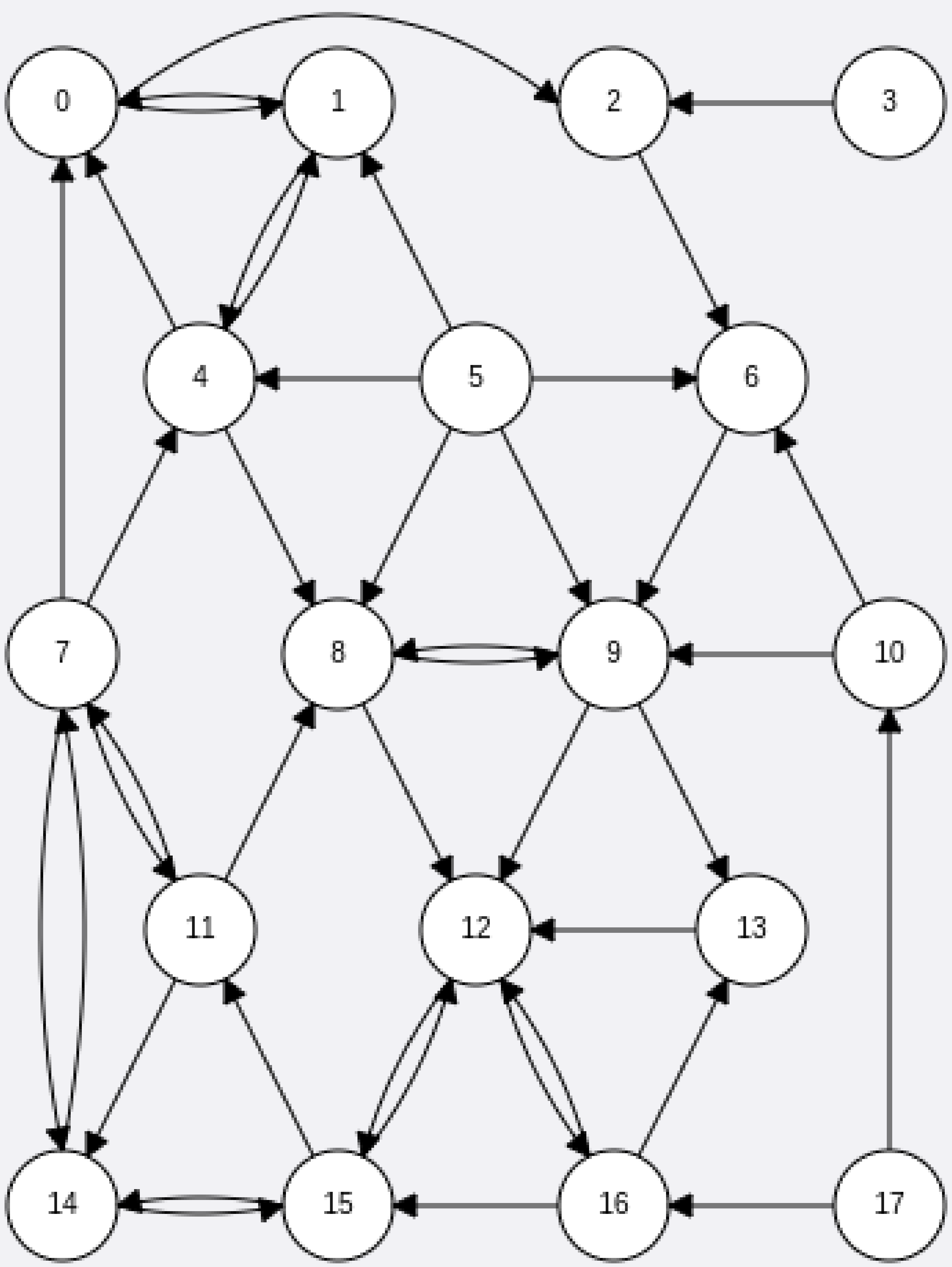


Exercise 3.7

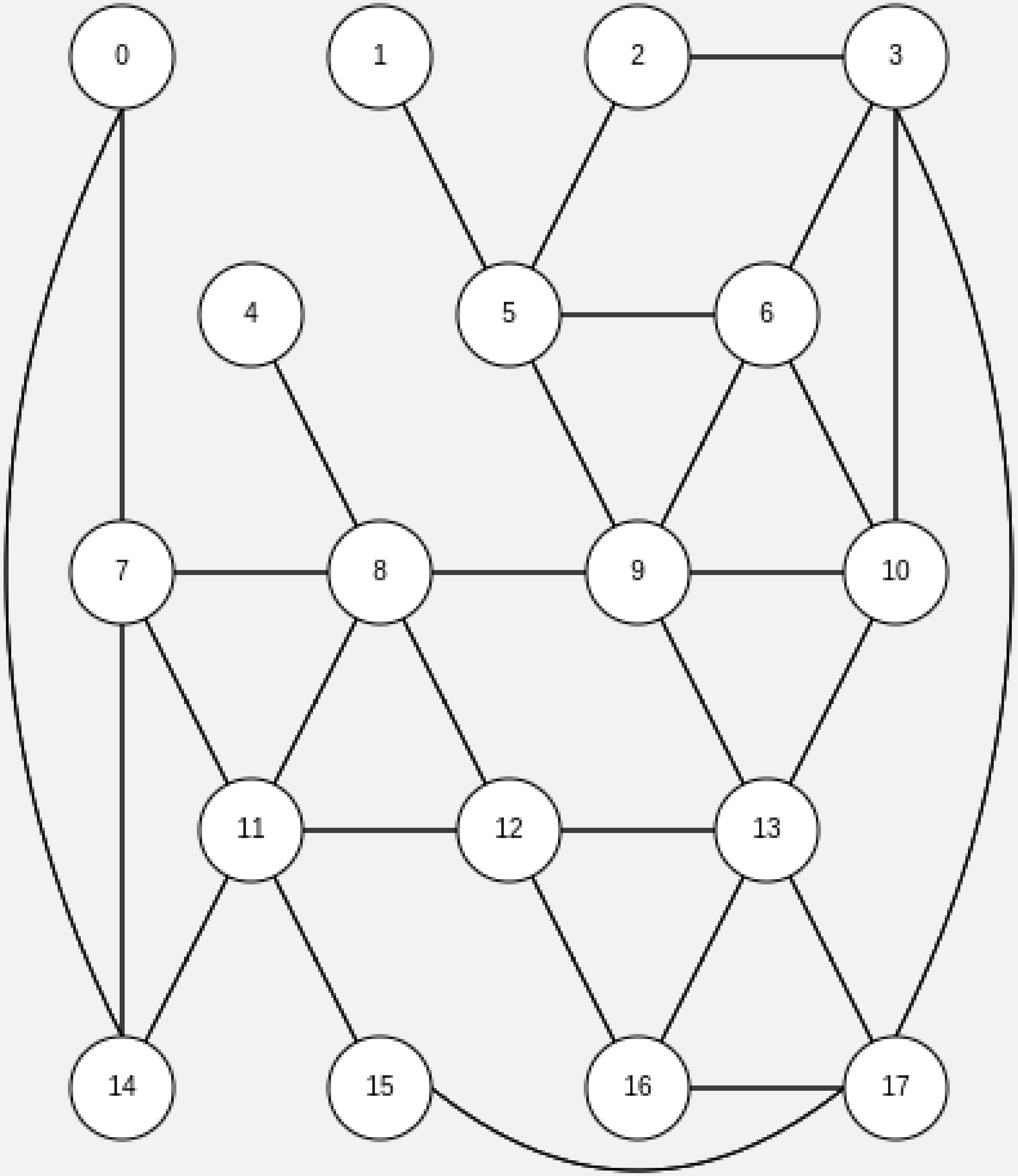


Exercise 3.7 - Solution

- **BFS:** 0 1 2 4 6 8 9 12 13 15 16 11 14 7
- **DFS:** 0 1 4 8 9 12 15 11 7 14 16 13 2 6

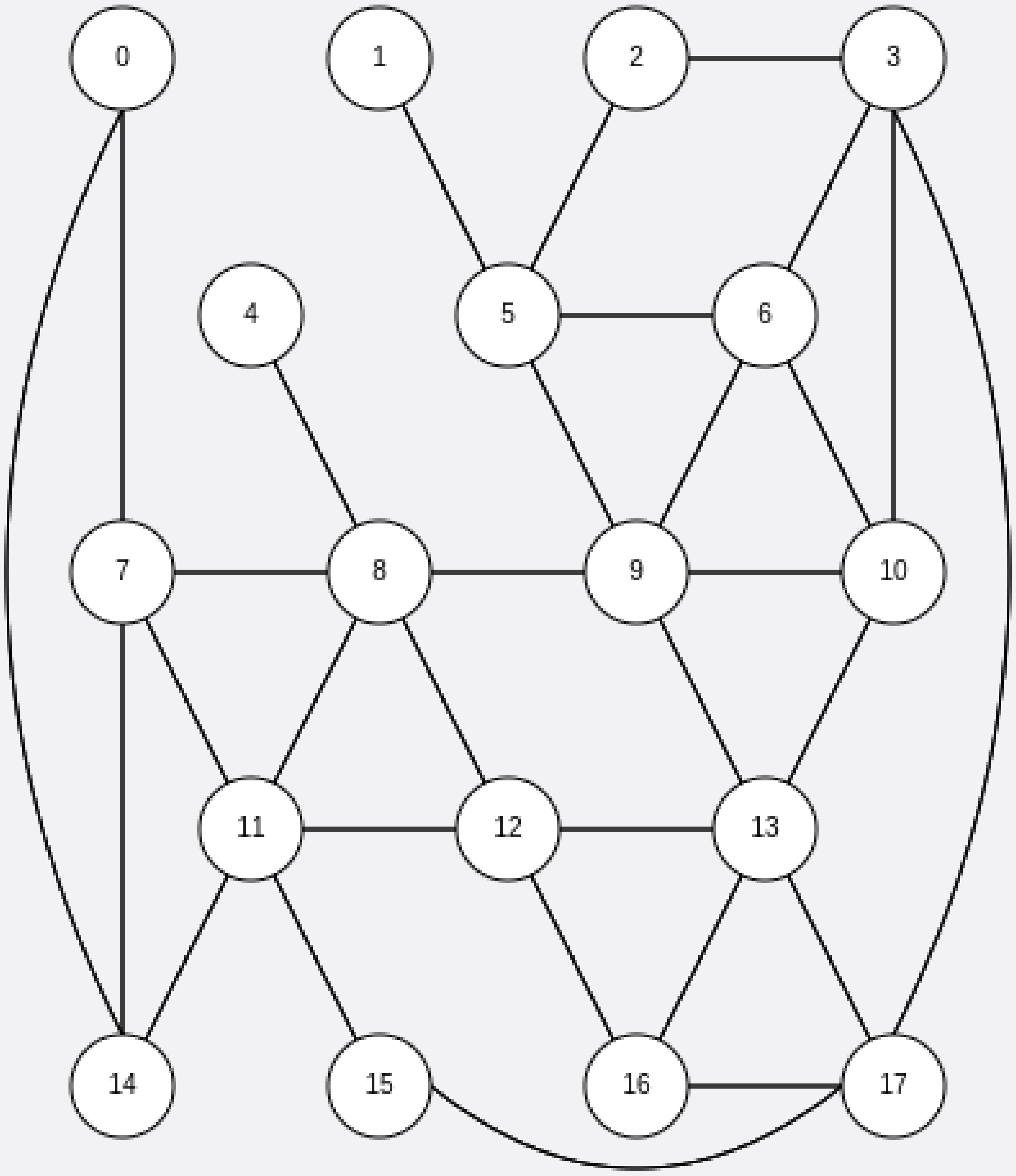


Exercise 3.8



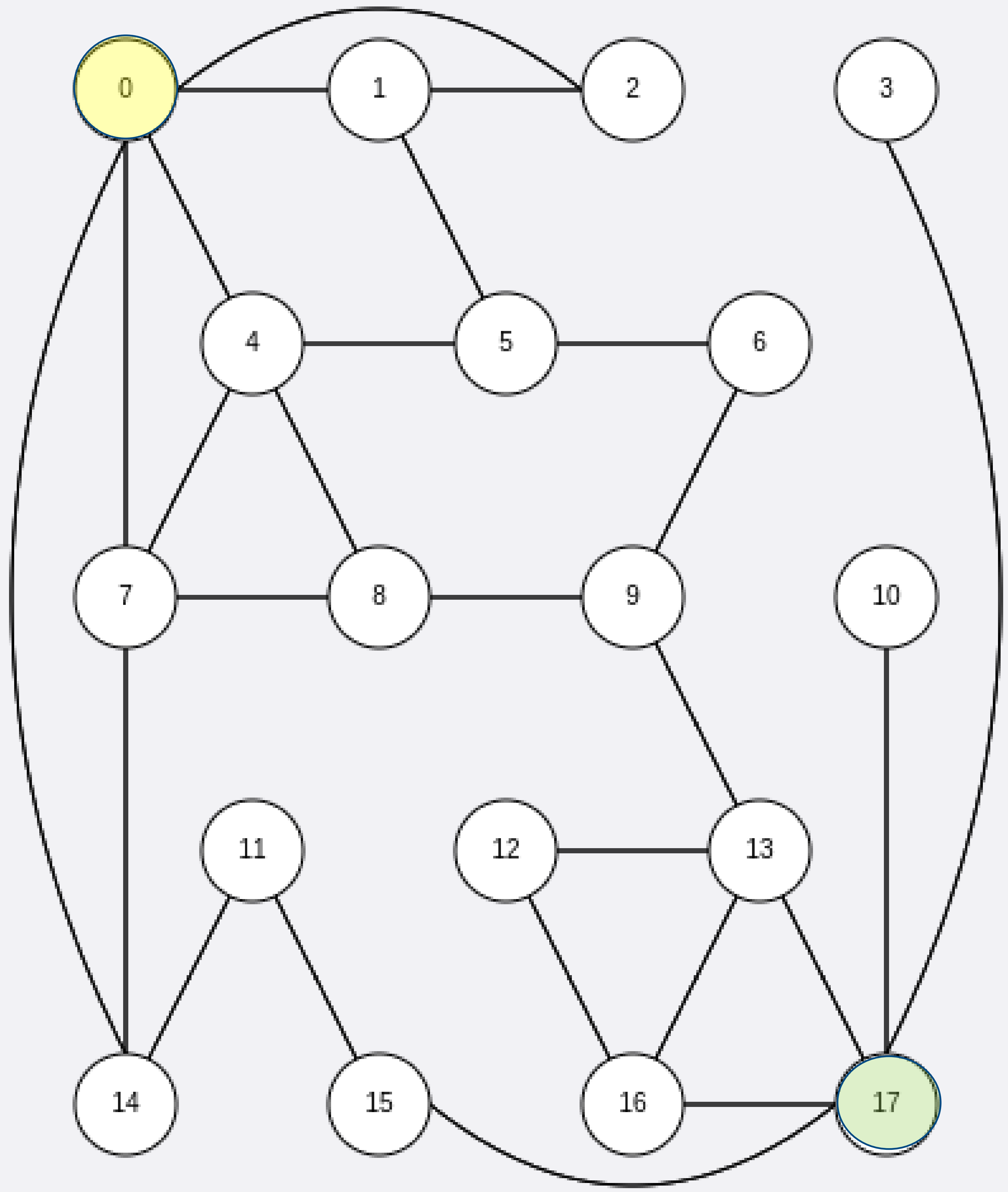
Exercise 3.8 - Solution

- **BFS:** 0 7 14 8 11 4 9 12 15 5 6 10 13 16 17 1 2 3
- **DFS:** 0 7 8 4 9 5 1 2 3 6 10 13 12 11 14 15 17 16



Exercise 3.9

- Apply both the **iterative deepening depth-first search** and the **bidirectional search** for reaching the goal (N-17) from the start (N-0)



Exercise 3.9 - Solution

- In order to avoid misunderstanding and to do not create confusion, we apply the algorithm as it is explained in the book without considering possible variants.

- **Iterative deepening**

  d0 = {0}
  d1 = {0,1,2,4,7,14}
  d2 = {0,1,2,4,7,14,5,8,11}
  d3 = {0,1,2,4,7,14,5,8,11,6,9,15}
  d4 = {0,1,2,4,7,14,5,8,11,6,9,15,13,**17**}

Exercise 3.9 - Solution

- In order to avoid misunderstanding and to do not create confusion, we apply the algorithm as it is explained in the book without considering possible variants.

- **Bidirectional search (by applying breadth-first)**
  Step0 = {0} {17}
  Step1 = {0,1,2,4,7,14} {17,3,10,13,15,16}
  Step2 = {0,1,2,4,7,14,5,8,**11**} {17, 3,10,13,15,16,9,12,**11**}

- **Bidirectional search (by applying depth-first)**
  Step0 = {0} {17}
  Step1 = {0,1} {17,3}
  Step2 = {0,1,2} {17,3,10}
  Step3 = {0,1,2,5} {17,3,10,13}
  Step4 = {0,1,2,5,4} {17,3,10,13,9}
  Step5 = {0,1,2,5,4,7} {17,3,10,13,9,6}
  Step6 = {0,1,2,**5**,4,7,8} {17,3,10,13,9,6,**5**}