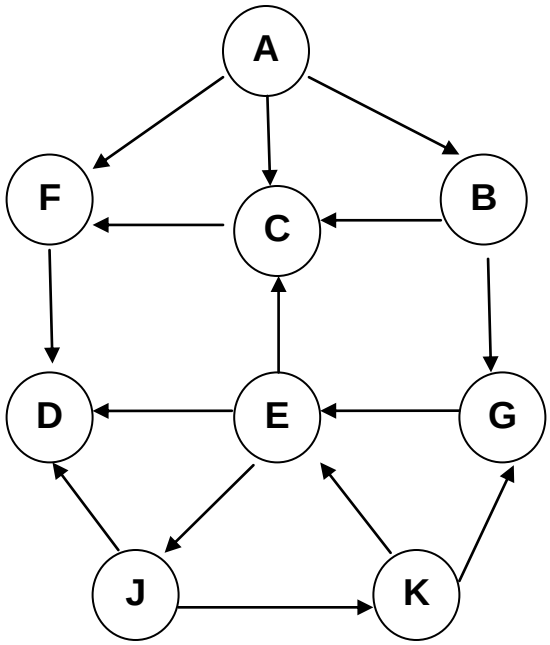
GRAPH TRAVERSALS

DEPTH FIRST SEARCH

1. Starting from vertex v
2. Mark v as marked
3. Select u as an unmarked node adjacent to v
4. If no u, quit
5. If u, begin depth first search from u
6. When search from u quits, select another node from v
7. Similar to pre-order tree traversal

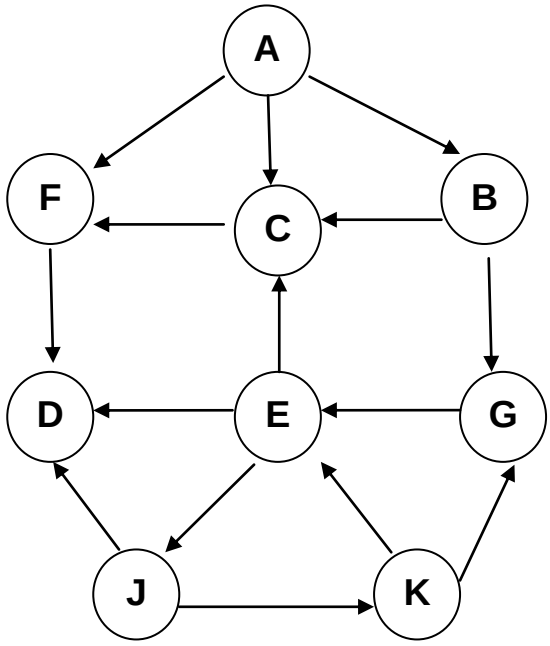
BREADTH FIRST SEARCH

1. Starting from node v
2. Identify all nodes adjacent to v
3. Add these to the set
4. Determine set of unvisited nodes which are adjacent to this set
5. Add these to the set
6. Continue until no new nodes are encountered



Depth First Search - DFS (Maintained by Stack)

Visited Vertex	Stack	DFS
Visit A	A	A
Visit F	A F	A F
Visit D	A F D	A F D
Pop D, F	A	A F D
Visit C	A C	A F D C
Pop C	A	A F D C
Visit B	A B	A F D C B
Visit G	A B G	A F D C B G
Visit E	A B G E	A F D C B G E
Visit J	A B G E J	A F D C B G E J
Visit K	A B G E J K	A F D C B G E J K
Pop K, J, E, G, B, A	-	**A F D C B G E J K**



Breadth First Search - BFS (Maintained by Queue)

Visited Vertex	Queue	BFS
Visit A	A	A
Remove A	-	A
Visit F	F	A F
Visit C	F C	A F C
Visit B	F C B	A F C B
Remove F	C B	A F C B
Visit D	C B D	A F C B D
Remove C	B D	A F C B D
Remove B	D	A F C B D
Visit G	D G	A F C B D G
Remove D	G	A F C B D G
Remove G	-	A F C B D G
Visit E	E	A F C B D G E
Remove E	-	A F C B D G E
Visit J	J	A F C B D G E J
Remove J	-	A F C B D G E J
Visit K	K	A F C B D G E J K
Remove K	-	**A F C B D G E J K**