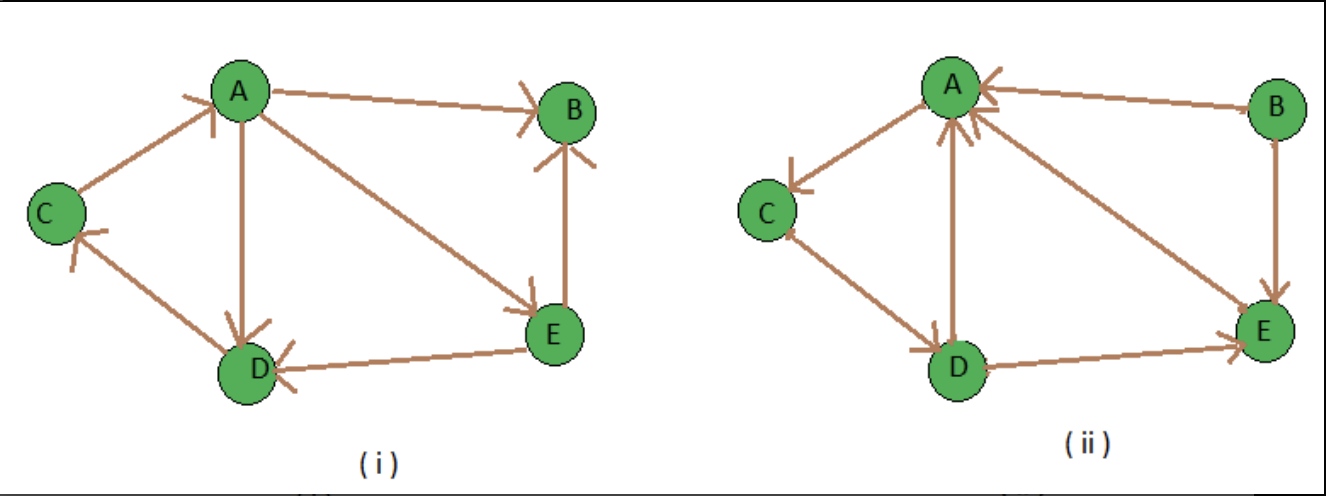
GRAPHS (almost last thing to learn in DSA)

1. If there is an undirected path between a and b then it is from both sides.
2. There is no need of circular link. There can be a graph without a cycle or we can say open cycle.
3. There can be no cycle in a closed graph in case of directed graph.
4. To check if the cycle is present then check if we can reach the starting point again if we traverse through.
5. Trees are the restricted types of graphs, just with some more rules. Every tree will always be a graph but not all graphs will be trees. [Linked List](https://www.geeksforgeeks.org/data-structures/linked-list/), [Trees](https://www.geeksforgeeks.org/binary-tree-data-structure/), and [Heaps](https://www.geeksforgeeks.org/heap-data-structure/) all are special cases of graphs. **Trees**: A connected graph with no cycles.
6. Degree of a graph is the number of edges attached to the node.
7. Smallest graph possible is the graph with single node and no vertex.
8. Connected Graph is the graph in which from one node we can visit any other node.
9. **Complete Graphs:** A graph in which each vertex is connected to every other vertex.
10. Parallel Edges: If two vertices are connected with more than one edge then such edges are called parallel edges that are many routes but one destination.
11. Transpose of a directed graph G is another directed graph on the same set of vertices with all of the edges reversed compared to the orientation of the corresponding edges in G

The Time complexity of BFS is O(V + E) when Adjacency List is used and O(V^2) when Adjacency Matrix is used, where V stands for vertices and E stands for edges

1. 