	Queue	Visited List
1	→ Add starting vertex, i.e. ***BOS*** 's neighbors in queue to get: **Queue =** [JFK, MIA, SFO] → And ***BOS*** becomes the current vertex in the Visited List	[BOS*]
2	→ There are three neighbors of the current vertex, i.e. ***BOS*** → For all neighbors of current vertex, i.e. ***BOS*** , repeat the process: ❶ Dequeue ***JFK*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***JFK*** 's neighbors in the queue → As all the neighbors have not been traversed, ***BOS*** remains the current vertex **Queue** = [MIA, SFO, BOS, DFW, MIA, SFO] ❷ Dequeue ***MIA*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***MIA*** 's neighbors in the queue → As all the neighbors have not been traversed, ***BOS*** remains the current vertex **Queue** = [SFO, BOS, DFW, MIA, SFO, DFW, LAX] ❸ Dequeue ***SFO*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***SFO*** 's neighbors in the queue → As all the neighbors have not been traversed, ***BOS*** remains the current vertex **Queue** = [BOS, DFW, MIA, SFO, DFW, LAX, LAX] → Now, as all neighbors of ***BOS*** have been traversed, so ***JFK*** becomes the current vertex	❶ [BOS*, JFK] ❷ [BOS*, JFK, MIA] ❸ [BOS*, JFK, MIA, SFO] → [BOS, JFK*, MIA, SFO]
3	→ There are four neighbors of the current vertex, i.e. ***JFK*** → For all neighbors of current vertex, i.e. ***JFK*** , repeat the process: ❶ Dequeue ***BOS*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***JFK*** remains the current vertex **Queue** = [DFW, MIA, SFO, DFW, LAX, LAX] ❷ Dequeue ***DFW*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***DFW*** 's neighbors in the queue → As all the neighbors have not been traversed, ***JFK*** remains the current vertex **Queue** = [MIA, SFO, DFW, LAX, LAX , LAX, ORD, SFO] ❸ Dequeue ***MIA*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***JFK*** remains the current vertex **Queue** = [SFO, DFW, LAX, LAX , LAX, ORD, SFO] ❹ Dequeue ***SFO*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***JFK*** remains the current vertex **Queue** = [DFW, LAX, LAX , LAX, ORD, SFO] → Now, as all neighbors of ***JFK*** have been traversed, so ***MIA*** becomes the current vertex	❶ [BOS, JFK*, MIA, SFO] ❷ [BOS, JFK*, MIA, SFO, DFW] ❸ [BOS, JFK*, MIA, SFO, DFW] ❹ [BOS, JFK*, MIA, SFO, DFW] → [BOS, JFK, MIA*, SFO, DFW]

4	→ There are two neighbors of the current vertex, i.e. ***MIA*** → For all neighbors of current vertex, i.e. ***MIA*** , repeat the process: ❶ Dequeue ***DFW*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***MIA*** remains the current vertex **Queue** = [LAX, LAX, LAX, ORD, SFO] ❷ Dequeue ***LAX*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***LAX*** 's neighbors in the queue → As all the neighbors have not been traversed, ***MIA*** remains the current vertex **Queue** = [LAX, LAX, ORD, SFO, ORD] → Now, as all neighbors of ***MIA*** have been traversed, so ***SFO*** becomes the current vertex	❶ [BOS, JFK, MIA*, SFO, DFW] ❷ [BOS, JFK, MIA*, SFO, DFW, LAX] → [BOS, JFK, MIA, SFO*, DFW, LAX]
5	→ There is one neighbor of the current vertex, i.e. ***SFO*** → For this neighbor of current vertex, i.e. ***SFO*** , repeat the process: ❶ Dequeue ***LAX*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***SFO*** remains the current vertex **Queue** = [LAX, ORD, SFO, ORD] → Now, as all neighbors of ***SFO*** have been traversed, so ***DFW*** becomes the current vertex	❶ [BOS, JFK, MIA, SFO*, DFW, LAX] → [BOS, JFK, MIA, SFO, DFW*, LAX]
6	→ There are three neighbors of the current vertex, i.e. ***DFW*** → For all neighbors of current vertex, i.e. ***DFW*** , repeat the process: ❶ Dequeue ***LAX*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***DFW*** remains the current vertex **Queue** = [ORD, SFO, ORD] ❷ Dequeue ***ORD*** , and check if already visited → It is <u>not</u> in the list, so mark it as visted by adding to the Visited List → Add ***ORD*** 's neighbors in the queue → As all the neighbors have not been traversed, ***DFW*** remains the current vertex **Queue** = [SFO, ORD, DFW, MIA] ❸ Dequeue ***SFO*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***DFW*** remains the current vertex **Queue** = [ORD, DFW, MIA] → Now, as all neighbors of ***DFW*** have been traversed, so ***LAX*** becomes the current vertex	❶ [BOS, JFK, MIA, SFO, DFW*, LAX] ❷ [BOS, JFK, MIA, SFO, DFW*, LAX, ORD] ❸ [BOS, JFK, MIA, SFO, DFW*, LAX, ORD] → [BOS, JFK, MIA, SFO, DFW, LAX*, ORD]
7	→ There is one neighbor of the current vertex, i.e. ***LAX*** → For this neighbor of current vertex, i.e. ***LAX*** , repeat the process: ❶ Dequeue ***ORD*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***LAX*** remains the current vertex **Queue** = [DFW, MIA] → Now, as all neighbors of ***LAX*** have been traversed, so ***ORD*** becomes the current vertex	❶ [BOS, JFK, MIA, SFO, DFW, LAX*, ORD] → [BOS, JFK, MIA, SFO, DFW, LAX, ORD*]

8	→ There are two neighbors of the current vertex, i.e. ***ORD*** → For all neighbors of current vertex, i.e. ***ORD*** , repeat the process: ❶ Dequeue ***DFW*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***ORD*** remains the current vertex **Queue** = [MIA] ❷ Dequeue ***MIA*** , and check if already visited → It is <u>already</u> in the list, so do nothing other than dequeueing → ***ORD*** remains the current vertex **Queue** = [ ] → The queue is empty and all the vertices have been traversed, so the traversal ends here, & the final order of visited list is: **[BOS, JFK, MIA, SFO, DFW, LAX, ORD]**	❶ [BOS, JFK, MIA, SFO, DFW, LAX, ORD*] ❷ [BOS, JFK, MIA, SFO, DFW, LAX, ORD*] → [BOS, JFK, MIA, SFO, DFW, LAX, ORD]