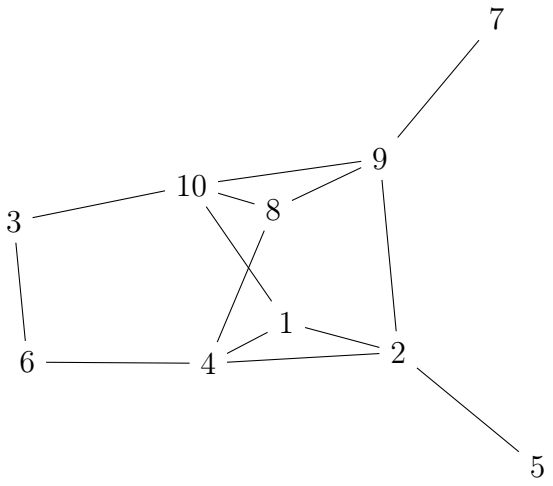
Use the adjacency lists on the left to draw the graph on the right:

1	2, 4, 10
2	1, 4, 5, 9
3	6, 10
4	1, 2, 6, 8
5	2
6	3, 4
7	9
8	4, 9, 10
9	2, 7, 8, 10
10	1, 3, 8, 9



The progress of breadth-first search, starting with 1. Try to use the adjacency lists, not the picture. Follow the order of each adjacency list, e.g., 1's adjacency list is [2, 4, 10], so enqueue 2, then 4, then 10 in that order. Build a picture of the breadth-first tree as you go.

visit	enqueue, and add tree edges to
1	2, 4, 10
2	5, 9
4	6, 8
10	3
5	
9	7
6	
8	
3	
7	

Suppose now the adjancency list of 1 is [10, 2, 4] instead of [2, 4, 10]. The progress of breadth-first search, starting with 1 (build the tree as you go):

visit	enqueue, and add tree edges to
1	10, 2, 4
10	3, 8, 9
2	5
4	6
3	
8	
9	7
5	
6	
7	

The point is that changing the order in an adjacency list affects the order of visits and the breadth-first tree.