**Breadth first search (without queue)**

1. First check the tree.
2. The tree is written using dictionary.
3. Start visiting the nodes by alphabetically until it reach its goal.
4. Which nodes are visited append it to a list.
5. Check the sub-nodes of the node.
6. If node is visited then add its children to the next level.
7. For next level we have use (level = child).
8. When goal is reach print visited nodes and break the function.
9. At the end, print the order in which all nodes are visited.

**Breadth first search (with queue)**

1. The tree is using dictionary.
2. Start exploring the nodes by alphabetically.
3. Exploring the nodes until goal is reached.
4. Which nodes are visited make a list and append it to that list.
5. Check queue by using while loop.
6. If queue is not empty remove the first node.
7. If node is not visited make it visited.
8. Add the sub-nodes to the end of queue
9. If it reach its goal print result and exit the loop.