

Prompt	Scratchpad	Our Solution(s)	Video Explanation	Run Code	Your Solutions	Run Code
--------	------------	-----------------	-------------------	----------	----------------	----------

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 using System.Collections.Generic;
4
5 public class Program {
6     public class Node {
7         public string name;
8         public List<Node> children = new List<Node>();
9     }
10
11     public Node(string name) {
12         this.name = name;
13     }
14
15     // O(v + e) time | O(v) space
16     public List<string> BreadthFirstSearch(List<string> array) {
17         Queue<Node> queue = new Queue<Node>();
18         queue.Enqueue(this);
19         while (queue.Count > 0) {
20             Node current = queue.Dequeue();
21             array.Add(current.name);
22             current.children.ForEach(o => queue.Enqueue(o));
23         }
24         return array;
25     }
26
27     public Node AddChild(string name) {
28         Node child = new Node(name);
29         children.Add(child);
30         return this;
31     }
32 }
33 }
```

Solution 1 Solution 2 Solution 3

```
1 using System.Collections.Generic;
2
3 public class Program {
4     // Do not edit the class below except
5     // for the BreadthFirstSearch method.
6     // Feel free to add new properties
7     // and methods to the class.
8     public class Node {
9         public string name;
10        public List<Node> children = new List<Node>();
11    }
12
13    public Node(string name) {
14        this.name = name;
15    }
16
17    public List<string> BreadthFirstSearch(List<string> array) {
18        // Write your code here.
19        return null;
20    }
21
22    public Node AddChild(string name) {
23        Node child = new Node(name);
24        children.Add(child);
25        return this;
26    }
27 }
28 }
```

Run or submit code when you're ready.