Our Solution(s) Run Code

Solution 1

```
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

using System.Collections.Generic;

public class Program {
  public class Node {
    public string name;
    public List<Node> children = new List<Node>();

    public Node(string name) {
      this.name = name;
    }

    // O(v + e) time | O(v) space
    public List<string> BreadthFirstSearch(List<string> array) {
      Queue<Node> queue = new Queue<Node>();
      queue.Enqueue(this);
      while (queue.Count > 0) {
        Node current = queue.Dequeue();
        array.Add(current.name);
        current.children.ForEach(o => queue.Enqueue(o));
      }
      return array;
    }

    public Node AddChild(string name) {
      Node child = new Node(name);
      children.Add(child);
      return this;
    }
  }
}

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```
using System.Collections.Generic;

public class Program {
  // Do not edit the class below except
  // for the BreadthFirstSearch method.
  // Feel free to add new properties
  // and methods to the class.
  public class Node {
    public string name;
    public List<Node> children = new List<Node>();

    public Node(string name) {
      this.name = name;
    }

    public List<string> BreadthFirstSearch(List<string> array) {
      // Write your code here.
      return null;
    }

    public Node AddChild(string name) {
      Node child = new Node(name);
      children.Add(child);
      return this;
    }
  }
}

```

Custom Output Raw Output Submit Code

Run or submit code when you're ready.