Our Solution(s)   Run Code   Your Solutions   Run Code

Solution 1

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

class Program {
    class Node {
        var name: String
        var children: [Node]

        init(name: String) {
            self.name = name
            children = []
        }

        func addChild(name: String) -> Node {
            let childNode = Node(name: name)
            children.append(childNode)

            return self
        }

        // O(v + e) time | O(v) space
        func breadthFirstSearch(array: inout [String]) -> [Strin
            var queue = [self]

            while queue.count > 0 {
                let currentNode = queue.removeFirst()
                array.append(currentNode.name)

                for child in currentNode.children {
                    queue.append(child)
                }
            }

            return array
```

Solution 1   Solution 2   Solution 3

```swift
class Program {
    class Node {
        var name: String
        var children: [Node]

        init(name: String) {
            self.name = name
            children = []
        }

        func addChild(name: String) -> Node {
            let childNode = Node(name: name)
            children.append(childNode)

            return self
        }

        func breadthFirstSearch(array: inout [String]) -> [Stri
            // Write your code here.
            return []
        }
    }
}
```

Our Tests   Custom Output   Submit Code

Run or submit code when you're ready.