Solution 3

Solution 2

21

23

24

Run Code

Your Solutions

Solution 1

14рх

Run Code

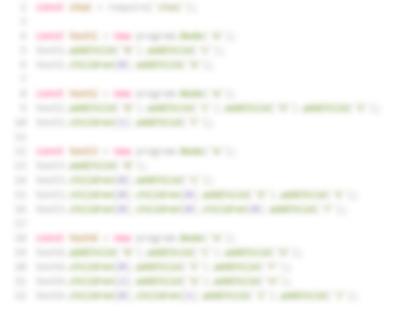
```
Solution 1
 1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
 3 class Node {
     constructor(name) {
       this.name = name;
 6
       this.children = [];
 7
 8
9
     addChild(name) {
10
       this.children.push(new Node(name));
11
       return this;
12
13
14
     // O(v + e) time | O(v) space
15
     breadthFirstSearch(array) {
16
      const queue = [this];
17
       while (queue.length > 0) {
18
        const current = queue.shift();
19
         array.push(current.name);
20
         for (const child of current.children) {
```

queue.push(child);

return array;

28 exports.Node = Node;

```
1 // Do not edit the class below except
 2 // for the breadthFirstSearch method.
 3 // Feel free to add new properties
 ^4\, // and methods to the class.
 5 class Node {
     constructor(name) {
 6
       this.name = name;
        this.children = [];
 9
10
11
      addChild(name) {
12
        this.children.push(new Node(name));
13
        return this;
14
15
16
      breadthFirstSearch(array) {
17
       // Write your code here.
18
19
20
21 // Do not edit the line below.
22 exports.Node = Node;
23
```



Run or submit code when you're ready.