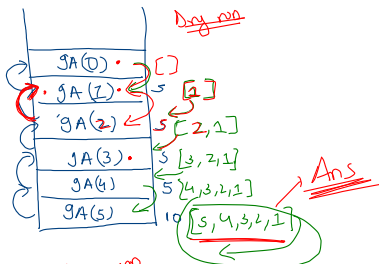


```

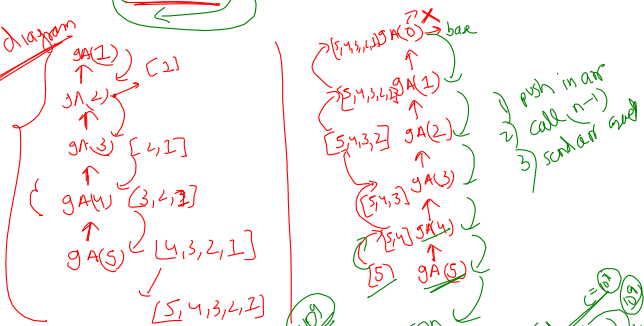
1 function getArray(num) {
2   if (num === 0) {
3     return [1];
4   }
5   let arr = getArray(num-1);
6   arr.unshift(num);
7   return arr;
8 }
9
10 let arr = getArray(5);
11 console.log(arr);

```

Deep recursion



Tree diagram



Complexity analysis:

- Time complexity:  $O(n)$
- Space complexity:  $O(n)$
- Recurrence relation:  $T(n) = T(n-1) + O(1)$
- Base case:  $T(0) = O(1)$
- Final result:  $T(n) = O(n)$

Space complexity:  $O(n)$

