AlgoMonster DSA Quiz

Study online at https://quizlet.com/_fbyr3d

1. **Which type of traversal does breadth first search do?:** Level-order traversal
2. **Which algorithm is best for finding the shortest distance between two points in an unweighted graph?:** Breadth-First Search
Technically, either depth-first search or breadth-first search can be used find the shortest distance two points in an unweighted graph.
However, in a large graph, depth-first search may go too far in the wrong direction, whereas breadth-first search will traverse the closest points first, and therefore be more efficient.
3. **A person thinks of a number between 1 and 1000. You may ask any number questions to them, provided that the question can be answered with either "yes" or "no".**
What is the minimum number of questions you needed to ask so that you are guaranteed to know the number that the person is thinking?: Correct Answer:10
A possible strategy is that each time, you divide the possible number into two distinct groups, and ask the question in a way so that you will know which group the answer belongs to. This way, you are guaranteed to know the answer within ceil(log2(1000)) == 10 questions.
This is the basic principle of binary search.
4. **What is the best way of checking if an element exists in an unsorted array once in terms of time complexity? Select the best that applies.:** Linear Search
5. **What's the output of running the following function using the following tree as input? (1 2 3 6)**

def serialize(root): 2 res = [] 3 def dfs(root): 4 if not root: 5 res.append('x') 6 return 7 res.append(root.val) 8 dfs(root.left) 9 dfs(root.right) 10 dfs(root) 11 return ' '.join(res)

function serialize(root) { 2 let res = []; 3 serialize_dfs(root, res); 4 return res.join(" "); 5} 6 7function serialize_dfs(root, res) { 8 if (!root) { 9 res.push("x"); 10 return; 11 } 12 res.push(root.val); 13 serialize_dfs(root.left, res); 14 serialize_dfs(root.right, res); 15} 16: Correct Answer: 1 2 3 x x x 6 x x
The code traverse the tree depth-first and prints the node value (x if node is null).
6. **How many ways can you arrange the three letters A, B and C?:** 6
The number of permutations between three letters is given by factorial of 3, ie. 3! = 3 × 2 × 1 = 6. The permutations are: ABC, ACB, BAC, BCA, CAB, CBA.
We can list all the permutations using backtracking.
7. **Consider the classic dynamic programming of fibonacci numbers, what is the recurrence relation?:** dp[i] = dp[i - 1] + dp[i - 2]

8. **What is an advantages of top-down dynamic programming vs bottom-up dynamic programming?:** Correct Answer:
Order of computer subproblems does not matter
9. **Is the following code DFS or BFS?**
1void search(Node root) { 2 if (!root) return; 3 visit(root); 4 root.visited = true; 5 for (Node node in root.adjacent) { 6 if (!node.visited) { 7 search(node); 8 } 9 } 10}: Depth First Search
10. **Which of the following uses divide and conquer strategy?:** Merge Sort
11. **How does quick sort divide the problem into subproblems?:** Correct Answer:
Divide the array into two based on whether an element is smaller than an arbitrary value
12. **Which of the following array represent a max heap?:** Correct Answer:
20 12 16 1 2 3 4
13. **A heap is a ...?:** Correct Answer:Tree
A heap is a tree with special "heap properties" - almost complete and each node's value is smaller/larger than its parent's value (min/max heap).
14. **What does the following code do?**

def f(arr1, arr2): 2 i, j = 0, 0 3 new_arr = [] 4 while i < len(arr1) and j < len(arr2): 5 if arr1[i] < arr2[j]: 6 new_arr.append(arr1[i]) 7 i += 1 8 else: 9 new_arr.append(arr2[j]) 10 j += 1 11 new_arr.extend(arr1[i:]) 12 new_arr.extend(arr2[j:]) 13 return new_arr

function f(arr1, arr2) { 2 let i = 0, j = 0; 3 let newArr = []; 4 5 while (i < arr1.length && j < arr2.length) { 6 if (arr1[i] < arr2[j]) { 7 newArr.push(arr1[i]); 8 i++; 9 } else { 10 newArr.push(arr2[j]); 11 j++; 12 } 13 } 14 15 while (i < arr1.length) { 16 newArr.push(arr1[i]); 17 i++; 18 } 19 20 while (j < arr2.length) { 21 newArr.push(arr2[j]); 22 j++; 23 } 24 25 return newArr; 26}: Correct Answer:
Merge two sorted arrays.
15. **Which two pointer techniques do you use to check if a string is a palindrome?:** Your Answer:
Two pointers moving in opposite direction