Data Structures and Algorithms (17ECSC204)

100 Questions Challenge Returns

Semester: III **Division: D** **Year: 2018-19**

While you see the hundred,
And you think of hundred,
In the time and space,
You are away from hundred!

Set A

[2 Points each. 5 Points for creative ones]

1. A maze runner does not know that maze he is running has no exit. Which traversal will make him realize this faster? DFS or BFS?
2. How are BST and AVL related?
3. Whom do we credit the invention of algorithms?
4. Why wasn't BST allowed to enter inside a party but whereas AVL and 2-3 Trees were allowed?
5. Who is in friends with Warshall's? Dijkstra's or Floyds?
6. Why was the root of the tree scared?
7. Why are single rotations of an AVL tree prouder than double rotations?
8. 2-3…. what comes next?
9. Who is more organized? Inorder? Preorder? Or Postorder?
10. Which is the best sorting algorithm that you know of, that the sorting algorithm does not know of?
11. If you were to design a T-shirt quote for Kruskal's, then what would be it?
12. What would Kruskal's algorithm purchase in the algorithm's market?
13. Why isn't Prim's algorithm celebrating No-Shave November?
14. If a DFS was run for every iteration of DFS, what would you finally get?
15. A level order traversal wasn't happy at all. What day was it?
16. A DFS post and pre order numbering was used on a DFS forest. The end number turned out to be 23. Anything weird?
17. If question number 17 was missing, who would be given the task to find it?
18. Why doesn't sorting lie?
19. A string search was initiated on an infinite text pattern. Where do you think the searcher has now reached?
20. Who is the head of algorithms family? Why?

Set B

[5 Points Each]

Define the following terms:

21. Tree	22. Graph	23. Binary Tree	24. Binary Search Tree
25. Ternary tree	26. Almost complete binary tree	27. N-ary tree	28. Heap
29. AVL Tree	30. Complete binary tree	31. 2-3 Tree	32. Decision Tree
33. Spanning Tree	34. Minimum Spanning Tree	35. Sorting	36. Bubble Sort
37. Strictly Binary tree	38. Insertion Sort	39. Selection Sort	40. Merge Sort
41. Quick Sort	42. Bottom Up Heap	43. Search	44. Heap Sort
45. Orders of Growth	46. Level of a Tree	47. Recurrence relations	48. Time Efficiency
49. Shortest path	50. Edge Relaxation		

Set C

[10 Points Each]

51. Why is sorting essential? Highlight its importance
52. What is the relation between binary search and binary search tree
53. Write a note on analysis framework and explain Asymptotic notations

54. Explain Master Theorem
55. Write a Note on application of trees
56. Write the mathematical analysis of towers of Hanoi problem
57. How is divide and conquer different from decrease and conquer
58. Elaborate transform and conquer. What does it mean?
59. Why is brute force an important technique? How can it be improved?
60. Write a short note on Dynamic programming and greedy technique.

Set D

[10 Points for each sub-question, each task]

61. For the numbers: 35, 72, 98, 56, 33, 89, 62
a. Create a Binary search tree, AVL tree and 2-3 tree
b. Write down the table populating the number of comparisons required to search each element in all the three cases

62. Construct a BST, 2-3 tree and AVL tree for the given inputs:

a. 5, 7, 9, 11, 34, 25, 67 b. 7, 6, 5, 4, 3, 2, 1 c. T E C H N O L O G Y
d. 89, 21, 13, 17, 96, 45, 37, 41, 83, 67 e. 88, 77, 44, 66, 99, 22, 11, 55 f. N U M B E R S

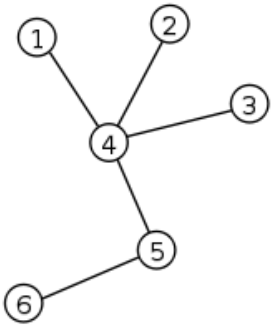
63. Trace the numbers: 67, 34, 98, 66, 38, 74, 56, 21 for:

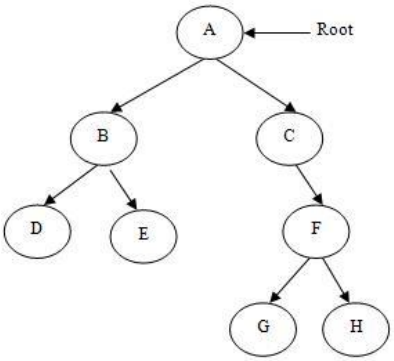
a. Bubble sort b. Insertion sort c. Selection Sort
d. Merge Sort e. Quick Sort f. Heap sort

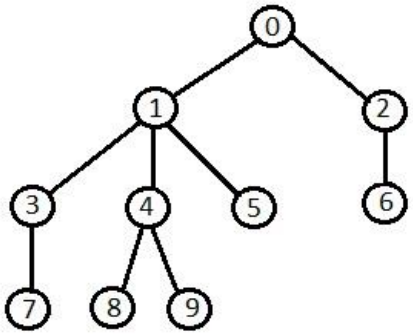
63. Write the efficiency analysis of following algorithms:

a. Bubble sort b. Insertion sort c. Selection Sort
d. Merge Sort e. Quick Sort f. Heap sort

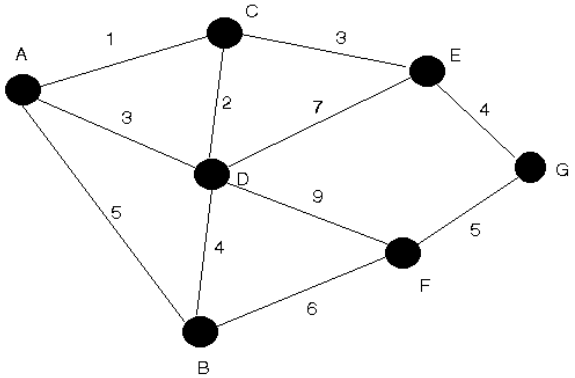
64. Apply DFS and BFS on the following Trees:

a. 

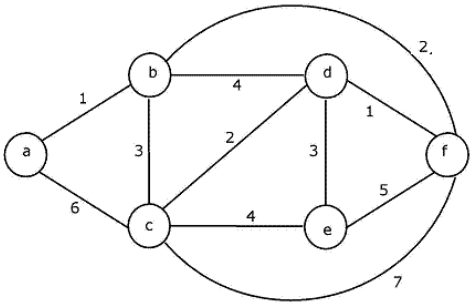
b. 

c. 

65. For the graph given below, iterate for Prim's and Kruskal's Algorithms:



66. Apply Dijkstra's algorithm on the following graph:



67. For the graph given in question number 66, show the iteration wise tracing using the data structures for Dijkstra's algorithm. Consider vertex 'd' as the source vertex.

68. For the graph given in question 66, apply Floyd's algorithm. Convert all the weights to 1 and then apply the Warshall's algorithm.

69. Consider the Quicksort algorithm. Suppose there is a procedure for finding a pivot element which splits the list into two sub-lists each of which contains at least one-fifth of the elements. Let T(n) be the number of comparisons required to sort n elements. Then write the recurrence relation for the said description.

70. Consider the following algorithm:

```
ALGORITHM Min1(A[]0… n-1)
if n = 1 return A[0]
else temp ← Min1(A[0… n-2])
   if temp <= A[n-1] return temp
   else return A[n-1]
```

a. What does the algorithm compute?
b. Which is the Basic Operation?
c. Set up a recurrence relation and solve it.
d. Which class of problem does the algorithm belong?
e. Can you make the algorithm efficient?

Set E
[20 Points each]

71. Which of the following is not O(n^2)?
a. (15^10) * n + 12099 b. n^1.98 c. n^3 / (sqrt(n)) d. (2^20) * n

72. Which of the given options provides the increasing order of asymptotic complexity of functions f1, f2, f3 and f4?
f1(n) = 2^n
f2(n) = n^(3/2)
f3(n) = nlogn
f4(n) = n^logn)
a. f3, f2, f1, f4 b. f2, f3, f1, f4 c. f3, f2, f4, f1 d. f2, f3, f4, f1

73. What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search?
a. N b. NlogN c. N^2 d.N(logN)^2

74. What is the time complexity of the below function?

```
void fun(int n, int arr[]) {
   int i = 0, j = 0;
   for(; i < n; ++i)
      while(j < n && arr[i] < arr[j])
```

```
        j++;
}
```
a. O(n^2) b. O(nlogn) c. O(n) d. O(n(logn)^2)

75. In a competition, four different functions are observed. All the functions use a single for loop and within the for loop, same set of statements are executed. Consider the following for loops:

a. for(i = 0; i < n; i++)
b. for(i = 0; i < n; i += 2)
c. for(i = 1; i < n; i *= 2)
d. for(i = n; i > -1; i /= 2)

If **n** is the size of input(positive), which function is most efficient(if the task to be performed is not an issue)?

a. a b. b c. c d. d

76. Consider the following segment of C-code:

```
 int j, n;
 j = 1;
 while (j <= n)
     j = j*2;
```

The number of comparisons made in the execution of the loop for any n > 0 is:

a. CEIL(logn) + 2 b. n c. FLOOR(logn) + 2 d. CEIL(logn)

77. Which of the following algorithms can be used to most efficiently determine the presence of a cycle in a given graph ?

a. Breadth First Search
b. Prim's Minimum Spanning Tree Algorithm
c. Depth First Search
d. Kruskal' Minimum Spanning Tree Algorithm

78. What are the appropriate data structures for following algorithms?

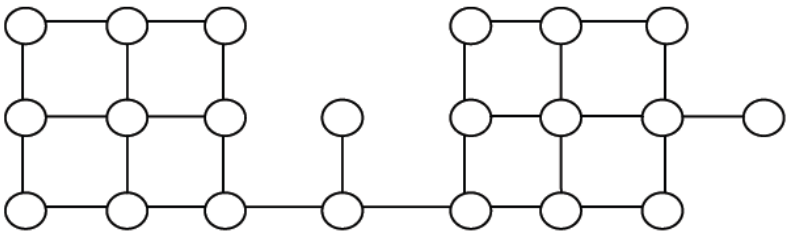
1) Breadth First Search
2) Depth First Search
3) Prim's Minimum Spanning Tree
4) Kruskal' Minimum Spanning Tree

a. 1) Stack	2) Queue	3) Priority Queue	4) Union Find
b. 1) Stack	2) Queue	3) Union Find	4) Priority Queue
c. 1) Queue	2) Stack	3) Priority Queue	4) Union Find
d. 1) Priority Queue	2) Queue	3) Stack	4) Union Find

79. Given two vertices in a graph s and t, which of the two traversals (BFS and DFS) can be used to find if there is path from s to t?

a. Only BFS
b. Only DFS
c. Both BFS and DFS
d. Neither BFS nor DFS

80. Suppose depth first search is executed on the graph below starting at some unknown vertex. Assume that a recursive call to visit a vertex is made only after first checking that the vertex has not been visited earlier. Then the maximum possible recursion depth is:



a. 17 b. 18 c. 19 d. 20

Set F

[50 Points each]

81. Opened up a dictionary to a page in the middle and start flipping through, looking for words you didn't know. You put each word that you didn't know at increasing indices in a huge array created in memory. When you reach the end of the dictionary, you start from the beginning and do the same thing until you reached the page you started at.
Now you have an array of words that are mostly alphabetical, except they start somewhere in the middle of the alphabet, reach the end, and then start from the beginning of the alphabet. In other words, this is an alphabetically ordered array that has been "rotated." For example:

```
char words[20][30] = {
"ptolemaic",
"retrograde",
"supplant",
"undulate",
"xenoepist",
"asymptote", // <-- rotates here!
"babka",
"banoffee",
"engender",
"karpatka",
"othellolagkage",
};
```

Write a function for finding the index of the "rotation point," which is where you started working from the beginning of the dictionary. Given example is just a sample and the array is actually huge. You need to think of an efficient way of achieving it.

82. You are given an array with duplicates. You have to sort the array with decreasing frequency of elements. If two elements have the same frequency, sort them by their actual value in increasing order.
Ex: [2 3 5 3 7 9 5 3 7] Output: [3 3 3 5 5 7 7 2 9]

83. You are given a rotated sorted array of size N. You have to search a given number into it.
Example: [4,6,8,14,90,-9,-2,0,3], Search -2.

84. Write a method to merge three sorted integer arrays into just one array

85. Given an array where each element is maximum +-k index away from its sorted position, find an algorithm to sort such array.

86. What is a best way to sort a linked list?

87. Given an array containing sequence of bits (0 or 1), you have to sort this array in the ascending order i.e. all 0' in first part of array followed by all 1's. The constraints is that you can swap only the adjacent elements in the array. Find the minimum number of swaps required to sort the given input array. Example: Given the array (0,0,1,0,1,0,1,1) the minimum number of swaps is 3.

88. Given a character array as input. Array contains only three types of characters 'R', 'G' and 'B'. Sort the array such that all 'R's comes before 'G's and all 'G's comes before 'B's. Constraint :- No extra space allowed(except O(1) space variables) and minimize the time complexity. You can only traverse the array once.

89. What is the best way to sort 1 million floating point numbers?

90. You are given an array which contains either 1 or 0 , and they are in sorted order Ex. a [] = { 1,1,1,1,0,0,0}. Devise an efficient method to count the number of 0's and 1's.

Set G

[100 Points each]

91. An artist had two lists A and B such that B was a permutation of A. Unfortunately, while transporting them from one exhibition to another, some numbers were left out of A. Can you find the missing numbers?

Notes: If a number occurs multiple times in the lists, you must ensure that the frequency of that number in both lists is the same. If that is not the case, then it is also a missing number. You have to print all the missing numbers in ascending order. Print each missing number once, even if it is missing multiple times. The difference between maximum and minimum number in B is less than or equal to 100.

Sample Input

10
203 204 205 206 207 208 203 204 205 206
13
203 204 204 205 206 207 205 208 203 206 205 206 204

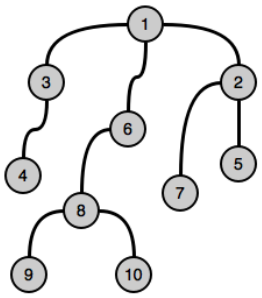
Sample Output:

204 205 206

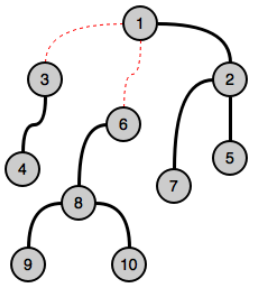
Explanation: 204 is present in both arrays. Its frequency in A is 2, while its frequency in B is 3. Similarly, 205 and 206 occur twice in A, but thrice in B. So, these three numbers are our output. The rest of the numbers have the same frequency in both lists.

92. You are given a tree (a simple connected graph with no cycles). The tree has N nodes numbered from 1 to N and is rooted at node. Find the maximum number of edges you can remove from the tree to get a forest such that each connected component of the forest contains an even number of nodes.

Original tree:

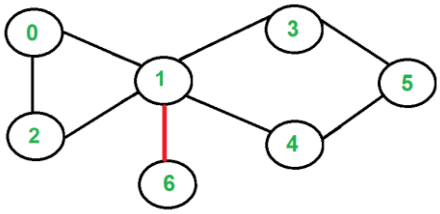


Decomposed tree:

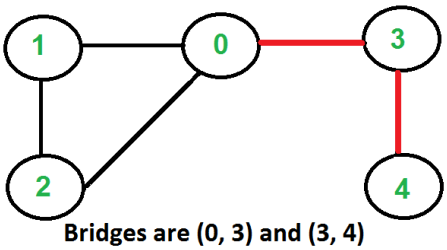


93. Given N integers, count the number of pairs of integers whose difference is K.

94. An edge in an undirected connected graph is a bridge iff removing it disconnects the graph. For a disconnected undirected graph, definition is similar, a bridge is an edge removing which increases number of connected components. Following are some example graphs with bridges:



Bridge is (1, 6)



Bridges are (0, 3) and (3, 4)

How to find all bridges in a given graph?

95. Given a Binary Tree, we need to print the bottom view from left to right.

```
          20
         /  \
        8    22
      /  \    \
     5    3    25
         /  \
        10   14
```

The output of above tree is: 5, 10, 3, 14, 25.

96. How many structurally unique BSTs are there for keys from 1..N?
For N = 3, there are 5 possible BSTs

```
 1          3     3     2     1
  \        /     /     / \     \
   3      2     1     1   3     2
  /      /       \             \
 2      1         2             3
```

97. The diameter of a tree (sometimes called the width) is the number of nodes on the longest path between two leaves in the tree. It might or not might pass through the root. Write a function to find diameter of the tree.

98. Devise an algorithm to detect a cycle in directed graph.

99. A mother vertex in a graph G = (V, E) is a vertex v such that all other vertices in G can be reached by a path from v. Write an algorithm to find the mother vertex.

100. Write an algorithm to check if the given binary tree is a binary search tree.

MAY THE FORCE BE WITH YOU

\- **PH**