DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF BARISHAL
Course Title: Data Structures
Course Code: CSE-1201
1st Year 2nd Semester
Session: 2022-23

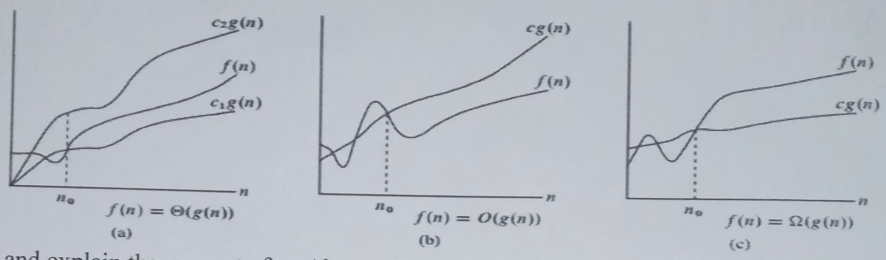
Time: 3 Hours **Marks:** 60

(Answer any FIVE questions)

1. a) Differentiate between data types and data structures. Provide examples of each. [3]

b) Why is knowledge of data structures important for computer science students? [3]

c) What do you mean by growth of function? Briefly describe the following (a), (b) and (c) of the figure. [3]



d) Define and explain the concept of an Abstract Data Type (ADT). How does it relate to data structures? [3]

2. a) Write down the algorithm for Insertion Sort and analyze its time complexity in best, average, and worst cases. [3]

b) Sort the following items using the Shell Sort algorithm: {7, 19, 24, 13, 31, 8, 82, 18, 44, 63, 5, 29} [5]

c) Using Quick Sort, sort the array A = {2, 8, 7, 1, 3, 5, 6, 4}. Describe the role of the partition function and how to select the pivot. [4]

3. a) Insert and delete elements in a Binary Search Tree (BST) using the numbers 33, 50, 45, 52, 12, 10. Draw the tree after each operation. [3]

b) Define a heap. Represent the heap as an array and perform the following operations: [3]

- Insert 25.
- Delete 86 and 11 from the heap.
- Sort the elements in descending order using Heap Sort.

c) Why do we need postfix and prefix notation over infix notation? Translate the infix notation $A*(B+D)/E-F*(G+H/K)$ to its equivalent postfix notation using stack. [4]

d) Compare and contrast BSTs and heaps. When would you prefer one over the other? [2]

4. a) Draw a graph from the adjacency list: adj(y) = [x], adj(x) = [z], adj(z) = [y,w], adj(w) = [x], adj(s) = [z,w], adj(v) = [s,w], adj(t) = [u,v], adj(u) = [v]. [2]

b) Create the adjacency and incidence matrices for the graph above. What are the advantages of using an adjacency matrix over an adjacency list for graph traversal? [3]

c) Identify the tree, back, forward, and cross edges for the graph. [4]

d) Write down the algorithms for the BFS algorithm. [3]

5. a) Write recursive formulas for: [4]

- Fibonacci sequence.
- GCD calculation.
- Tower of Hanoi problem.

b) What data structure is commonly used to implement recursion? Why? [2]

c) **What do you mean by disjoint set data structure?** Suppose that CONNECTED-COMPONENTS is run on the undirected graph G=(V, E), where V = { a, b, c, d, e, f, g, h, i, j, k} [4]

and the edges of E are processed in the order (d, i), (f, k), (g, i), (b, g), (a, h), (i, j), (d, k), (b, j), (d, f), (g, j), (a, e). **List the vertices in each connected component after each iteration of lines 3–5.**

```
CONNECTED-COMPONENTS(G)
1  for each vertex v ∈ G.V
2      MAKE-SET(v)
3  for each edge (u, v) ∈ G.E
4      if FIND-SET(u) ≠ FIND-SET(v)
5          UNION(u, v)
```

d) How can you differentiate hash table from the direct addressing table? [2]

6. a) Implement a queue using two stacks. Analyze the time complexity of enqueue and dequeue operations. [3]

b) For a circular queue with `FRONT = 2`, `REAR = 4`, and `QUEUE: _, A, C, D, _, _`, describe the state after: [3]

- Adding `F`.
- Deleting two elements.
- Adding `K, L, M`.
- Deleting two elements.

c) Describe how a priority queue can be used to implement: [3]

- A FIFO queue.
- A stack

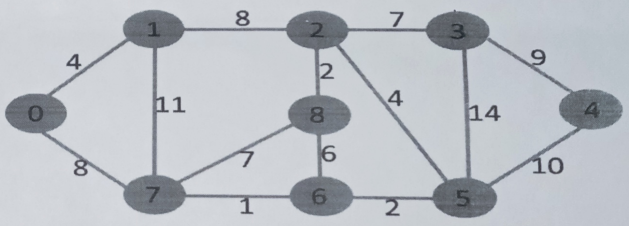
d) Define divide and conquer. A well-known problem in UVA- 374: Big Mod Problem, calculate $B^P \% M$ for large B, P and M using an efficient algorithm. Write down the program for calculating the solutions. [3]

7. a) Use a hash table of size 7 with separate chaining, linear probing, and double hashing to insert: `19, 26, 13, 48, 17`. Show the resulting tables. [3]

b) Explain how collision resolution techniques like chaining and double hashing affect performance. [3]

c) Solve the problems Longest Common Subsequence (LCS) using dynamic programming for strings `ABCBDAB` and `BDCAB`. Write the DP table and explain the solution. [3]

d) What do you mean by MST? Find the MST for the following graph using PRIM'S algorithm. Is prims a Greedy algorithm? [3]



8. a) Define divide and conquer. How does it apply to Merge Sort and Quick Sort [3]

b) Define complete binary tree. Traverse the given tree using In-order, Preorder and Post-order traversals. [3]

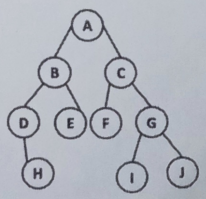
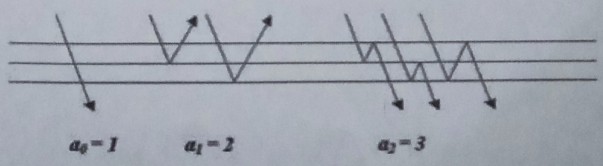


Figure for Q-8(b)

c) A well renowned UVA Problem: 10334, Suppose we put two panes of glass back-to-back. How many ways a_n are there for light rays to pass through or be reflected after changing direction n times? Following figure shows the situations when the value of n is 0, 1 and 2. Now your task is to write down the program for input 0, 1, 2: output will be 1, 2, and 3 re [3]

Figure for Q-8(a)(ii)



d) Working Modulo With $q = 11$, how many spurious hits does Rabin-Karp encounter when looking for pattern, P = 26 in text, T = 3141592653589793? Elaborate your answer. [3]