

Techno India NJR Institute of Technology



Course File

Data Structures & Algorithm (3CS4- 05)

Akhilesh Deep Arya
(Assistant Professor)
Department of CSE



RAJASTHAN TECHNICAL UNIVERSITY, KOTA

Syllabus

II Year-III Semester: B.Tech. Computer Science and Engineering

3CS4-05: Data Structures and Algorithms

Credit-3
3L+0T+0P

Max. Marks: 150 (IA:30, ETE:120)
End Term Exam: 3 Hours

SN	CONTENTS	Hours
1	Stacks: Basic Stack Operations, Representation of a Stack using Static Array and Dynamic Array, Multiple stack implementation using single array, Stack Applications: Reversing list, Factorial Calculation, Infix to postfix Transformation, Evaluating Arithmetic Expressions and Towers of Hanoi.	8
2	Queues: Basic Queue Operations, Representation of a Queue using array, Implementation of Queue Operations using Stack, Applications of Queues- Round Robin Algorithm. Circular Queues, DeQueue Priority Queues. Linked Lists: Introduction, single linked list, representation of a linked list in memory, Different Operations on a Single linked list, Reversing a single linked list, Advantages and disadvantages of single linked list, circular linked list, double linked list and Header linked list.	10
3	Searching Techniques: Sequential and binary search. Sorting Techniques: Basic concepts, Sorting by: bubble sort, Insertion sort, selection sort, quick sort, heap sort, merge sort, radix sort and counting sorting algorithms.	7
4	Trees: Definition of tree, Properties of tree, Binary Tree, Representation of Binary trees using arrays and linked lists, Operations on a Binary Tree, Binary Tree Traversals (recursive), Binary search tree, B-tree, B+ tree, AVL tree, Threaded binary tree.	7
5	Graphs: Basic concepts, Different representations of Graphs, Graph Traversals (BFS & DFS), Minimum Spanning Tree(Prims &Kruskal), Dijkstra's shortest path algorithms.Hashing: Hash function, Address calculation techniques, Common hashing functions, Collision resolution: Linear and Quadratic probing, Double hashing.	8
TOTAL		40

Course Overview:

Student will learn basics of DSA from this 40 hours course. They will be able to perform operations such as insertion, deletion and search, on linear and non linear data structures. Linear data structures covered under this course are array, stack, queue, double ended queue and linked list. In non linear data structure this course will cover trees and graphs. Student will learn and implement bubble, selection, insertion, quick and bucket sort algorithm using C programming language. This course will also cover the implementation of linear search, binary search and hashing techniques using C Language.

DSA is the basic requirement for the job role of software developer in the companies like TCS, Infosys etc. Most of the questions asked during the placement drive for the IT Company are created from this subject. Student should learn and develop problem solving abilities using DSA in order to get a good job in IT company.

Course Outcomes:

CO. NO.	Cognitive Level	Course Outcome
1	Synthesis	Student will be able to design algorithms and convert those algorithms into a C language code to perform push and pop operation on stack data structure. Students also develop an ability to perform recursion and apply them to the tower of Hanoi problem.
2	Synthesis	Student will be able to design algorithms and convert those algorithms into a C language code to perform enqueue, dequeue and traversing operation on queue and Linked list data structure. Student will also able to list the advantages and disadvantages of Linked List.
3	Application	Students will be able to write C code to implement Linear search, Binary Search, bubble sort, Insertion sort, selection sort, quick sort, heap sort, merge sort, radix sort and counting sort.
4	Application	Students will be able to write C programming code to create binary tree and implement pre, post and in order traversing on the tree data structure.
5	Application	Students will be able to write C programming code to implement Hashing. He should be able to perform breadth and depth first search operations on Graph data structure.

Prerequisites:

1. Fundamentals of C programming.
2. Students should be efficient in writing code using looping statements.
3. Students should be able to perform insertion and deletion operations on stack.
4. Students should be able to implement structures using pointers in C programming.

Course Outcome Mapping with Program Outcome:

Course Outcome	Program Outcomes (PO's)											
	CO. NO.	Domain Specific (PO)					Domain Independent (PO)					
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	3	2	2	1	-	-	1	-	-	1
CO2	2	2	3	2	2	1	-	-	1	-	-	1
CO3	2	2	3	2	2	1	-	-	1	-	-	1
CO4	3	2	3	2	2	1	-	-	1	-	-	1
CO5	3	2	3	2	2	1	-	-	1	-	-	1

1: Slight (Low) , 2: Moderate (Medium), 3: Substantial (High)

Course Coverage Module Wise:

Lect. No.	Unit	Topic
1	1	Stacks: Introduction and real world examples
2	1	Student should be able to write algorithm for stack push and pop operations
3	1	Student should be able to write functions in C programming to perform push, pop, peak and display operation on stack
4	1	Student should be able to write C program to reverse a string using stack data structure
5	1	Student should be able to write C program to check balanced parenthesizing using stack data structure
6	1	Student should be able to design an algorithm, also can write a C programming code to convert infix expression into postfix expression
7	1	Student should be able to write a C code to evaluate postfix expression using stack data structure
8	1	Student should write a recursive code using C programming to solve Towers of Hanoi, having 'n' number of rings
9	2	Queues: Introductions and real world examples
10	2	Student should be able to write algorithm for enqueue and dequeue operations on queue data structures
11	2	Students should be able to write C code to implement queue operations using stack.
12	2	Students should be able to write C code for Round Robin Algorithm using Queue data structure
13	2	Students should be able to write C code to implement enqueue and dequeue operations on double ended queue
14	2	Linked Lists: Introduction and real world examples
15	2	Student should be able to write C code for creating a singly linked list and to implement insertion and deletion operations on it
16	2	Student should be able to implement stacks using linked list

17	2	Student should be able to implement Queue using linked list
18	2	Student should be able to write C code to implement Circular Queue using linked list
19	2	Student should be able to write a C code to reverse a singly linked list without creating additional spaces
20	2	Student should be able write C code for creating a doubly linked list and to implement insertion and deletion operations on it
21	3	Searching Techniques: Sequential and binary search
22	3	Student should be able to write C code for linear search and binary search
23	3	Student should be able to write C code to implement bubble sort
24	3	Student should be able to write C code to implement selection sort and insertion sort
25	3	Student should be able to write C code to implement Quick sort considering pivot element at the end of the array
26	3	Student should be able to write C code to implement Heap sort using divide and conquer method
27	3	Student should be able to write C code to implement Merge sort using divide and conquer method
28	3	Student should be able to write C code for counting sort algorithm
29	4	Trees: Definition of tree, Properties of tree, array and linked representation of Tree
30	4	Student should be able to write C code to create a binary tree using linked representation
31	4	Student should be able to write recursive functions in C programming for the pre, post and, inorder traversal of binary tree
32	4	Student should be able to write C code to count num of nodes, num of leaf nodes and internal nodes in a binary tree
33	4	Student should be able to write C code to implement Binary Tree
34	4	Students should be able to list the difference between a binary tree and a B+ tree
35	4	Student should be able to write the advantages of balanced tree also list the advantages of balanced tree
36	4	Students should be able to create a threaded binary tree using
37	4	Student should be able to balance an unbalanced tree using the concepts and rotations explained in the AVL tree
38	5	Graphs: Basic concepts, Different representations of Graphs
39	5	Student should be able to write Graph Traversals (BFS & DFS) algorithm and can perform it on the given set of elements
40	5	Student should be able to write Minimum Spanning Tree (Prims & Kruskal) algorithm and implement the same on the given set of elements
41	5	Student should be able to write Dijkstra's shortest path algorithm and drive the shortest path from the graph
42	5	Student should be able to write C code to implement Hashing

TEXT/REFERENCE BOOKS

1. Data structures, Schaum Series, S. Lipshutz
2. Data Structures in C, Reema Thareja, Oxford University Press
3. An introduction to data structures with applications By Jean-Paul Tremblay, P. G. Sorenson, TMH
4. Data Structures in C, Tanenbaum, Pearson

Course Level Problems (Test Items):

CO.NO.	Problem description
1	A. Write an algorithm to perform push and pop operation on stack. B. Write a program to reverse a string using stack. C. Write a recursive function in C to solve tower of Hanoi problem.
2	A. Write a C program to perform enqueue and dequeue operations in queue. B. Write a C program to implement round robin scheduling algorithm. C. Write an algorithm to insert and delete elements from various locations of Linked List
3	A. Write an algorithm of binary search. B. Write a C program to implement Bubble sort. C. Write a C program to implement Selection Sort. D. Write an algorithm of quick sort.
4	A. Write a C program to implement binary tree. B. Write a C program to traverse binary tree. C. Explain the balance factor of tree and draw a balanced AVL tree using the data given.
5	A. Write an algorithm for breadth first search and depth first search. B. Find minimum spanning tree using Prim's algorithm C. Write a C program to implement the concept of Hashing.

Assessment Methodology:

1. Online quiz on kahoot after every module completion.
2. Practical exam in lab where they have to write code on C compiler for the given problem statement. (Once in a week)
3. Assignments one from each unit.
4. Midterm subjective paper where they have to write algorithms to perform different operations on different data structures as mentioned in the modules. (Twice during the semester)
5. Final paper at the end of the semester subjective.

Teaching and Learning resources unit-wise:

Unit-1

Stack implementation and application of stack.

Video Tutorials:

1. https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s
2. <https://youtu.be/SBy-OVSOOjQ>
3. <https://youtu.be/wKi0y-9RNRs>
4. <https://youtu.be/dO9HBypj6I8>
5. <https://youtu.be/77OVgLg8KZ0>
6. <https://youtu.be/8C1hMq2UIt0>

Theory concepts: <https://www.geeksforgeeks.org/stack-data-structure/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/stack-gq/>

NPTEL Course:

<https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F>

Unit-2

A. Queue implementation and applications of queue

Video Tutorials:

- <https://youtu.be/qXRZWcaL-10>
- https://youtu.be/b7pb_8JsrpM
- <https://youtu.be/7cQ2jP5Kesc>
- <https://youtu.be/3Pv644T5Z3o>

Theory concepts: <https://www.geeksforgeeks.org/queue-data-structure/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/queue-gq/>

B. Linked list implementation and various operations on linked list

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/data-structures/linked-list/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/linked-list-gq/>

NPTEL Course:

<https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F>

Unit-3

A. Linear and binary search

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/searching-algorithms/>

Sample Quiz: <https://www.geeksforgeeks.org/algorithms-gq/searching-gq/>

B. Sorting techniques

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/sorting-algorithms/>

Sample Quiz: <https://www.geeksforgeeks.org/algorithms-gq/searching-and-sorting-gq/>

NPTEL Course:

<https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F>

Unit-4

A. Fundamentals of Trees

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/binary-tree-data-structure/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/binary-trees-gq/>

B. Binary search tree implementation

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/binary-search-tree-data-structure/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/binary-search-trees-gq/>

C. B+ Trees

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/introduction-of-b-tree/>

Sample Quiz: <https://www.geeksforgeeks.org/introduction-of-b-tree/>

D. AVL Tree

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/avl-tree-set-1-insertion/>

Sample Quiz: <https://www.geeksforgeeks.org/avl-tree-set-2-deletion/>

NPTEL Course:

<https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F>

Unit-5

A. Graph

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/graph-gq/>

B. Hashing

Video Tutorials: https://www.youtube.com/watch?v=a32_3hiZCU0&t=73s

Theory concepts: <https://www.geeksforgeeks.org/hashing-data-structure/>

Sample Quiz: <https://www.geeksforgeeks.org/data-structure-gq/hash-gq/>

NPTEL Course:

<https://www.youtube.com/watch?v=zWg7U0OEAoE&list=PLBF3763AF2E1C572F>

Previous Year Question Papers:

Total No. of Questions: 25

Total No. of Pages: 03

Roll No. _____

B.Tech. III-Sem. (Main/Back) Exam Jan. 2019

Computer Science Engineering
3CSU02 Data Structures and Algorithms

3EU3022

Time: 3 Hours

Maximum Marks: 100
Min. Passing marks: 33

Instructions to candidates: -

PART A : Short answer questions (up to 25 words) 10 x 2 marks = 20 marks.
All ten questions are compulsory.

PART B : Analytical Problem Solving questions (up to 100 words) 6 x 5 marks = 30 marks. Candidates have to answer six questions out of eight.

PART C : Descriptive Analytical/Problem solving questions 5 x 10 marks = 50 marks. Candidates have to answer five questions out of seven.

PART A

Q.1 Why we need to do algorithm analysis ?

Q.2 Discuss BST.

Q.3 Write the name of an algorithm which can be used as a single source single destination shortest path algorithm.

Q.4 Give Comparison between tree and graph ?

Q.5 Define Spanning tree. What is MST ?

Q.6 Which data structures are used for BFS and DFS of a graph ?

Q.7 What are linear and non-linear data structures ?

Q.8 What is linked list? What are its types?

Q.9 Explain the advantages of Binary search over linear search ?

It access data randomly

Q.10 How is an array different from Linked list ?

PART B

Q.1 Write the following infix expressions in their postfix and prefix forms:

(a) $D-B+C$

(b) $(A+B)*C-D*F+C$

Q.2 What is queue? How it is different from stack and how is it implemented ?

Q.3 Create lexically ordered Binary Search Tree for the following :-

JAN, FEB, MAR, APR, MAY, JUNE, JULY, AUG, SEPT, OCT, NOV, DEC.

Q.4 Write the essential differences between complete binary tree and strict binary tree?

Q.5 Calculate the address of the element $A[3,2]$ using row major order for an array $A[1..5, 1..5]$ of elements. It is stored at location 2033 and the size of each element is 3 Bytes. <http://www.rtuonline.com>

Q.6 How insertion and selection sorts are different ? Explain.

Q.7 What is the value of the following postfix expression :

$6, 8, 2, 4, +, -, * - \> \Delta ?$

Q.8 What is asymptotic analysis of an algorithm ? What are asymptotic notations ?

PART C

Q.1 Create the AVL tree.

21, 26, 30, 9, 4, 14, 28, 18, 15, 10, 2, 3, 7.

Q.2 Write an algorithms for inserting a node and deleting a node from a doubly linked list ? What are the advantages of doubly linked list over singly linked list ?

Q.3 Define AVL tree ? Discuss the term "Balance factor". Explain the various rotations of AVL tree?

Q.4 Analyze the running time for merge sort algorithm Argye upon its wrost case. best case and average case running time.

Q.5 Write shote note (any two) :-

(i) Heap sort

(ii) B- tree

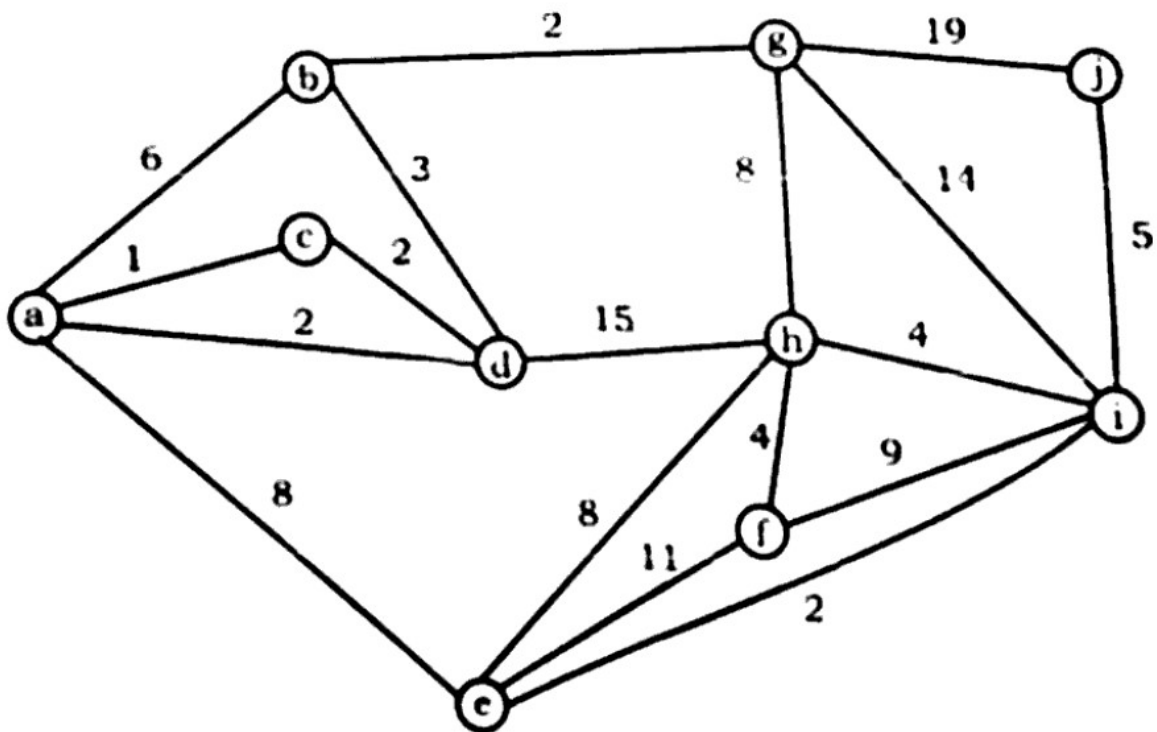
(iii) Tree traversal techniues

Q.6 What are the various ways to repreasant a graph? Find the following two for the graph given below in Q.7 :

(i) Adjacency list representation

(ii) Adjacency matric representation

Q.7 Using prim's and kruskal's algorithm , find the minimum spanning tree for the following graph ? What is the weight of a minimum spanning tree of the following graph?



3E1138

Roll No.

Total No of Pages: 4

3E1138

B. Tech. III - Sem. (Main / Back) Exam., Dec. 2019

PCC Computer Science & Engineering

3CS4-05 Data Structures and Algorithms

CS, IT

Time: 3 Hours

Maximum Marks: 120

Instructions to Candidates:

Attempt all ten questions from Part A, five questions out of seven questions from Part B and four questions out of five from Part C.

Schematic diagrams must be shown wherever necessary. Any data you feel missing may suitably be assumed and stated clearly. Units of quantities used /calculated must be stated clearly.

*Use of following supporting material is permitted during examination.
(Mentioned in form No. 205)*

1. NIL

2. NIL

PART - A

(Answer should be given up to 25 words only)

[10×2=20]

All questions are compulsory

- Q.1 Define data structure. Mention any two applications of data structures.
- Q.2 Mention the purpose of B⁺ - Trees.
- Q.3 What is the difference between internal sorting and external sorting?
- Q.4 What is meant by abstract data type?
- Q.5 What are the applications of stack?
- Q.6 What do you mean by circular linked list?

Q.7 Compare graph and tree.

Q.8 Differentiate between linear and non-linear data structure.

Q.9 What is a dequeue?

Q.10 Define Hash function.

PART – B

(Analytical/Problem solving questions)

[5×8=40]

Attempt any five questions

Q.1 Difference between linear queue and circular queue. Also write the advantage and disadvantage of circular queue. [8]

Q.2 What do you mean by tower of Hanoi problem? Explain with suitable example. [8]

Q.3 Convert following expressions in its equivalent post fix expressions – [8]

(i) $A * (B + C * D) + E$

(ii) $A * B ^ C + D$

Q.4 Define Binary Search Tree. Write algorithm to implement insertion operation on Binary search tree. [8]

Q.5 The in – order & pre – order traversal sequence of nodes in a binary tree are given below:

In-order: E A C K F H D B G

Pre-order: F A E K C D H G B

Draw the binary tree. [8]

Q.6 What is a priority queue? How can it be implemented? Explain an application of priority queue. [8]

Q.7 What is a Threaded Binary Tree? Explain the advantages of using a threaded binary tree. [8]

PART – C

(Descriptive/Analytical/Problem Solving/Design Questions)

[4×15=60]

Attempt any four questions

Q.1 Create the linked list to represent the following polynomials –

[15]

Ⓐ $5x^5 + 4x^4 + 6x^2 - 4$
 $8x^6 + 4x^4 + 3x^3 + 2x^2 + x$

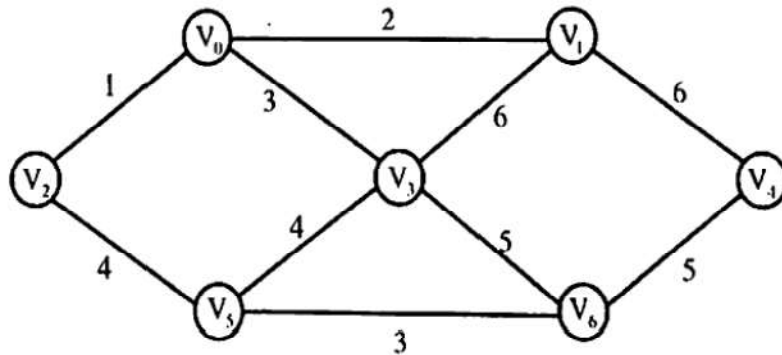
Write a function add () to add these polynomials and print the resultant linked list.

Q.2 Define a B-Tree. What are the application of B-Tree? Draw a B-Tree of order 4 (four) by insertion of the following keys in order :

Z, U, A, I, W, L, P, X, C, J, D, M, T, B, Q, E, H, S, K, N, R, G, Y, F, O, V. [15]

Q.3 What is sorting? Write an algorithm to sort the real number using insertion sort and selection sort. What is the time complexity for both selection and insertion sort? [15]

Q.4 (a) Define the spanning tree. Write the Prim's algorithm to find the minimum cost spanning tree of the following: <http://www.rtuonline.com> [8]



Ⓐ (b) Describe the Dijkstra's algorithm for finding shortest path with help of suitable example. [7]

- Q.5 (a) What is AVL tree? Explain the balancing methods of AVL tree with an example. [8]
- (b) What do you mean by hashing and collision? Discuss the advantages and disadvantages of hashing over other searching techniques. [7]
-