Indraprastha College for Women
University of Delhi

Work Plan for ODD SEMESTER – 2026

Course Name:	B.Sc.(H) Computer Science
Paper Title:	Algorithms and Advanced Data Structures
Unique Paper Code:	
Semester:	V
Faculty(s):	Dr. Manisha Bansal
Year:	2026

	Work Plan			
Week No.	Unit No.	Learning Objective	Reference No.	Topics to be Covered
28 Jul - 31Jul	III	More on Trees:	[4]	Ch 7 - 7.1.1 (B-Trees)
3 Aug - 8Aug	III	More on Trees:	[4]	Ch 7 - 7.1.1 (B-Trees)
10Aug - 15Aug	III	More on Trees:	[4]	7.1.8 (upto the derivation of expression for relation between height and number of nodes in a 2-4 tree (excluding insertion, deletion and related operations)
17Aug - 22Aug	III	More on Trees:	[4]	7.1.8 (upto the derivation of expression for relation between height and number of nodes in a 2-4 tree (excluding insertion, deletion and related operations)
24Aug - 29Aug	III	More on Graphs: Bellman Ford Algorithm,	[3]	Ch 6 - 6.8
31Aug - 5Sep	III	More on Graphs:,Union Find Data Structures - application Kruskal's algorithm	[3]	Ch 4 - 4.6
7Sep - 12Sep	II	String Matching: Brute Force, KMP algorithm	[1]	Ch 12 - 12.3.1, 12.3.3

14Sep - 19Sep	II	String Matching: Brute Force, KMP algorithm	[1]	Ch 12 - 12.3.1, 12.3.3
21Sep - 26Sep	II	Tries: Standard Tries, Compressed Tries, Suffix Tries, Search Engines	[1]	Ch 12 - 12.5
28Sep - 3Oct	II	Network Flows: Ford Fulkerson algorithm for the max flow problem	[3]	Ch 7 - 7.1
5Oct - 11Oct	IV	Randomization: Randomized Quicksort, Randomized Select	[2]	Ch 7 - 7.3, Ch 9 - 9.2
12Oct - 17Oct	IV	Randomization: Skip lists	[1]	Ch 9 - 9.4 (Intro, 9.4.1)
19Oct - 24Oct	Mid Semester Break			
26Oct - 31Oct	I	Advanced Analysis of Algorithms	[2]	Ch 16 - 16.1, 16.2
2Nov - 7Nov	I	Intractability	[3]	Ch 8 - 8.1, 8.3, 8.4 (upto theorem 8.12)
9Nov - 14Nov	I	Intractability	[3]	Ch 8 - 8.1, 8.3, 8.4 (upto theorem 8.12)
16Nov - 19Nov		Revision		

Unit	Contents/Syllabus
I	Unit 1 Advanced Analysis of Algorithms, Intractability Ch 16 - 16.1, 16.2. [2] Ch 8 - 8.1, 8.3, 8.4. [3] (upto theorem 8.12)

II	String Matching: Brute Force, KMP algorithm, Tries: Standard Tries, Compressed Tries, Suffix Tries, Search Engines Network Flows: Ford Fulkerson algorithm for the max flow problem Ch 12 - 12.3.1, 12.3.3, [1] Ch 12 - 12.5. [1] Ch 7 - 7.1. [3]
III	More on Trees: B Trees, 2-4 Trees, More on Graphs: Bellman Ford Algorithm, Union Find Data Structures - application Kruskal's algorithm Ch 7 - 7.1.1, 7.1.8 (upto the derivation of expression for relation between height and number of nodes in a 2- 4 tree (excluding insertion, deletion and related operations) [4] Ch 6 - 6.8. [3] Ch 4 - 4.6 [3]
IV	Randomization: Randomized Quicksort, Randomized Select, Skip ists Ch 7 - 7.3, Ch 9 - 9.2. [2] Ch 9 - 9.4 (Intro, 9.4.1). [1]

S. No.	Name of Authors/Books/Publishers
1.	Goodrich, M.T, Tamassia, R., & Mount, D. Data Structures and Algorithms Analysis in C++, 2nd edition, Wiley, 2011
2.	Cormen, T.H., Leiserson, C.E., Rivest, R. L., Stein C. Introduction to Algorithms, 4th edition, Prentice Hall of India, 2022.
3.	Kleinberg, J., Tardos, E. Algorithm Design, 1st edition, Pearson, 2013.
4.	Drozdek, A. Data Structures and Algorithms in C++, 4th edition, Cengage Learning. 2012.

	Paper Components		
Credits	**Lecture (L)**	**Tutorial (T)**	**Practical (P)**
4	**3**	**0**	**1**

Assessment Scheme

S.No.	**Component**	**Marking Scheme**	**Total Marks**
1	Internal Assessment		**30**
	• Assignment/Quiz/Project/ Presentation	12	
	• Class Test	12	

	• Attendance	6	
2.	Continuous Assessment (**Tutorial)**		NA
	• Activity 1		
	• Activity 2		
	• Attendance		
3.	Practical		40
	• Continuous Assessment	10	
	• End Term Written/Practical Exam	20	
	• Viva	10	
4.	End Semester Examination		90

Practical List (30 Hours)

1. Write a program to sort the elements of an array using Randomized Quick Sort (the program should report the number of comparisons).
2. Write a program to find the ith smallest element of an array using Randomized Select.
3. Write a program to determine the minimum spanning tree of a graph using Kruskal's algorithm.
4. Write a program to implement the Bellman-Ford algorithm to find the shortest paths from a given source node to all other nodes in a graph.
5. Write a program to implement a B-Tree.
6. Write a program to search a pattern in a given text using the KMP algorithm.
7. Write a program to implement a Suffix tree (bruteForceSuffixTrie method given in Ref. 4 at page 712).