A. Course Handout (Version 1.0) | Last updated on 6th Dec, 2024

Institute/School Name	Chitkara University Institute of Engineering and Technology		
Department Name	Department of Computer Science & Engineering		
Programme Name	Bachelor of Engineering (B.E.), Computer Science & Engineering		
Course Name	Data Structures using Object Oriented Programming		
Course Code	22CS013	Semester/Batch	4th /2022
L-T-P (Per Week)	3-0-4	Course Credits	05
Course Coordinator	Dr. Anuj Kumar Jain		

1. Objectives of the Course

Data structures play a central role in modern computer science. Data structures are essential building blocks in obtaining efficient algorithms. This course covers elementary data structures (dynamic arrays, heaps, balanced binary search trees, hash tables) and algorithmic approaches to solve classical problems (sorting, graph searching, dynamic programming). It Introduces the mathematical modeling of computational problems, as well as common algorithms, algorithmic paradigms, and data structures used to solve these problems.

The main objectives of the course are:

- To use object-oriented programming knowledge for solving real-world problem statements.
- To evaluate time-space complexity trade-offs for all categories of algorithms.
- To understand concepts of searching and sorting techniques.
- To understand basic concepts of stacks, queues, list, trees, and graphs.
- To understand writing algorithms and step-by-step approach in solving problems with the help of fundamental data structures.

CLO01	Understand the basics of data structure, complexity of algorithms, and the implementation of various operations on arrays and linked lists.
CLO02	Illustrate the concepts of stack and queue with their applications and apply recursion to solve certain problems.
CLO03	Persuade different searching, sorting, and hashing mechanisms with their comparisons.
CLO04	Understand, implement, and analyze graph data structure and apply it to real-world problems.
CLO05	Analyze different tree traversal techniques and understand various kinds of trees.

2. Course Learning Outcomes

After completion of the course, the student should be able to:

	Course Outcome	POs	CL*	KC**	Sessions
CLO01	Understand the basics of data structure, complexity of algorithms, and the implementation of various operations on arrays and linked lists.	PO1, PO2, PO3, PO4, PO9, PO11, PO12	K2	Factual Conceptual	**51**
CLO02	Illustrate the concepts of stack and queue with their applications and apply recursion to solve certain problems.	PO1, PO2, PO3, PO4, PO9, PO11, PO12	K3	Conceptual Procedural	**40**
CLO03	Persuade different searching, sorting, and hashing mechanisms with their comparisons.	PO1, PO2, PO3, PO4,	K3	Conceptual Procedural	**18**

		PO9, PO11, PO12			
CLO04	Understand, implement, and analyze graph data structure and apply it to real-world problems.	PO1, PO2, PO3, PO4, PO9, PO11, PO12	K4	Conceptual Metacognitive	**14**
CLO05	Analyze different tree traversal techniques and understand various kinds of trees.	PO1, PO2, PO3, PO4, PO9, PO11, PO12	K3	Conceptual	**17**
Total Contact Hours					**140**

Revised Bloom's Taxonomy Terminology
*Cognitive Level =CL
**Knowledge Categories = KC

LO-PO Mapping grid |Program outcomes (POs) are available as a part of the Academic Program Guide

Course Learning Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CLO1	H	H	H	M					L		L	H
CLO2	H	H	H	M					L		L	H
CLO3	H	H	H	H					M		M	H
CLO4	H	H	H	H					H		M	H
CLO5	H	H	H	H					H		M	H

H=High, M=Medium, L=Low

3. ERISE Grid Mapping

Feature Enablement	**Level(1-5, 5 being highest)**
Entrepreneurship	2
Research	5
Innovation	4
Skills	5
Employability	3

4. Recommended Books:

Text Books

B01: Data Structures Through C++, Yashavant Kanetkar, BPB Publications, 3rd Edition, 2019.
B02: Data Structure, Algorithms and Applications Using C++, S. Sahni, Sillicon Press, 2nd edition, 2005

Reference Books

B03: Introduction to Algorithms by Thomas H. Cormen, The MIT Pressm 3rd Edition, 2001

B04: Data Structures Using C and C++ by Langsam, Yedidyah, Tenenbaum, Aaron M., Pearson, 2nd edition, 2006

B05: Objects, Abstraction, Data Structures and Design Using C++, Elliot B. Koffman, Paul A. T. Wolfgang, Wiley Publications, First Edition 2006

E-Resources:

https://ndl.iitkgp.ac.in/

https://www.vlab.co.in/

5. **Other readings and relevant websites:**

Serial No	Link of Journals, Magazines, websites and Research Papers
1.	https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-096-introduction-to-c-january-iap-2011/lecture-notes/
2.	https://www.cs.bham.ac.uk/~jxb/DSA/dsa.pdf
3.	https://nptel.ac.in/courses/106/102/106102064/
4.	https://cse.iitkgp.ac.in/~pb/algo1-pb-101031.pdf
5.	https://ocw.mit.edu/courses/civil-and-environmental-engineering/1-204-computer-algorithms-in-systems-engineering-spring-2010/lecture-notes/MIT1_204S10_lec05.pdf
6.	http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.868.7428&rep=rep1&type=pdf

6. **Recommended Tools and Platforms**

Coding Blocks, Dev-C++

7. **Course Plan:**

Lecture Numbers	Topics	Books
1-4	Data Structures and Algorithms: Basic Terminology, Elementary Data Organization, Data Structures and Operations	B01-Chpater-1
5-9	Algorithm : Complexity, Time and Space & Complexity, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o)	B02-Chpater-3
10-14	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays.	B01-Chpater-2
15-18	Searching: Linear and Binary Search with their Complexity.	B01-Chpater-9
19-23	Character Arrays, Strings, Declaration and Initialization of character array, Memory Representation, Basic Operations. Sorting a Char Array.	B06-Chpater-7
24-27	Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort.	B06-Chpater-10
28-35	Bitmasking, Bitwise operations in Bitmasking, STL (Standard Template Library), Components of the STL.	B06-Chpater-11
36-40	Recursion & Backtracking: Introduction Recursion and Recursive Function	B02-Chpater-1
41-45	Deep Diving into recursion, Implementation of Recursion on Subsets, Backtracking Algorithms.	B02-Chpater-1
ST-1(Lectures 1-45)		
46-50	Preliminaries of Object-oriented programming using C++: keywords, identifiers, data types.	B02-Chpater-1
51-55	Decision control and looping statements, Functions.	B02-Chpater-1

56-59	Basics of objects and classes, Dynamic Memory Management & pointers,	B02-Chpater-1
60-63	Program structure of C++. Conversion of Algorithm into Program.	B02-Chpater-1
64-67	Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting from Linked List	B01-Chpater-3
68-70	Operations on Doubly Linked List, Circular linked List & its applications	B01-Chpater-3
71-73	Stacks: Array representation of Stacks, implementation of stack using linked list.	B01-Chpater-5
74-81	Applications: Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks	B01-Chpater-3
82-86	Queues: Representation as Array and Linked List.	B01-Chpater-6
87-90	Deques, Circular Queues, Priority Queues.	B01-Chpater-6
ST-2 (Lectures 46-90)		
91-98	Trees: Binary trees, complete binary trees, Binary Search Trees, Data structures for representing binary trees, Insertion, Deletion and Searching of Binary Search	B01-Chpater-7
99-107	Tree and their Implementation, Tree Traversal: preorder, In order, Post order and their algorithms, Their Implementation. B Trees.	B01-Chpater-7
108-117	Heaps, Difference between heap and Array, insertion and deletion in heap, Heap sort implementation and its applications.	B03-Chpater-6
118-120	Graphs: Basic terminology, directed and undirected graphs, notion of path.	B01-Chpater-8
121-125	Representation of graphs: edge list structures, adjacency list structures, adjacency matrix, Linked List representation of Graph.	B01-Chpater-8
126-128	Operations on Graph, Graph traversals: DFS, BFS and their implementation.	B01-Chpater-8
129-131	Hashmap, Hashing Techniques, Collision and its resolving. Trie Data Structure, Basic Operations on Trie.	B02-Chpater-10
ST-3 (Lectures 91-131)		
132-136	Introduction to Dynamic Programming, Standard problems on Dynamic Programming, One Dimensional Dynamic Programming.	B03-Chpater-15
137-140	Two Dimensional Dynamic Programming, Top DP Algorithms Greedy Algorithms	B03-Chpater-15
End Term Examination		

8. Delivery/Instructional Resources

Lecture Numbers	Topics	Web References	Audio-Video References

1-4	Data Structures and Algorithms: Basic Terminology, Elementary Data Organization, Data Structures and Operations	https://portal.abuad.edu.ng/lecturer/documents/1604393139CSC_207-slide1-introduction_and_terminologies.pptx	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-2-data-structures-and-dynamic-arrays/
5-9	Algorithm : Complexity, Time and Space & Complexity, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o)	https://ocw.mit.edu/courses/1-204-computer-algorithms-in-systems-engineering-spring-2010/8ee75d49f1cb9a947f1d3f15a2aa9e00_MIT1_204S10_lec05.pdf	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-1-algorithms-and-computation/
10-14	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays.	https://home.csulb.edu/~hill/ee444/Lectures%20-%20Deprecated/04%20C++%20Arrays%20Arduino.pdf	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-2-data-structures-and-dynamic-arrays/
15-18	Searching: Linear and Binary Search with their Complexity.	https://courses.cs.washington.edu/courses/cse143/12wi/lectures/01-13/05-binarysearch-complexity.pdf	https://www.youtube.com/watch?v=k4xVQhMERuQ
19-23	Character Arrays, Strings, Declaration and Initialization of character array, Memory Representation, Basic Operations. Sorting a Char Array.	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1132/handouts/08-C++-Strings.pdf	https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/resources/session-16-strings/
24-27	Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort.	https://ocw.mit.edu/courses/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/resources/mit6_0001f16_lec12/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-5-linear-sorting/
28-35	Bitmasking, Bitwise operations in Bitmasking, STL (Standard Template Library), Components of the STL.	https://faculty.cs.byu.edu/~rodham/cs240/lecture-notes/Lecture-25-STL/Lecture-25-STL.pdf	https://www.youtube.com/watch?v=BKBXM7ypQG0
36-40	Recursion & Backtracking: Introduction Recursion and Recursive Function	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1206/lectures/intro-to-recursion/	https://ocw.mit.edu/courses/6-00sc-introduction-to-computer-science-and-programming-spring-2011/resources/lecture-6-recursion/
41-45	Deep Diving into recursion, Implementation of Recursion on Subsets, Backtracking Algorithms.	https://web.mit.edu/6.005/www/fa15/classes/10-recursion/	https://www.youtube.com/watch?v=GOs07Kn2W1E

46-50	Preliminaries of Object-oriented programming using C++: keywords, identifiers, data types.	https://ocw.mit.edu/courses/6-088-introduction-to-c-memory-management-and-c-object-oriented-programming-january-iap-2010/resources/mit6_088iap10_lec04/	https://ocw.mit.edu/courses/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/resources/lecture-8-object-oriented-programming/
51-55	Decision control and looping statements, Functions.	https://www.cs.uic.edu/~jbell/CourseNotes/CPlus/Looping.html	https://www.youtube.com/watch?v=Y0TYYEhHvIU
56-59	Basics of objects and classes, Dynamic Memory Management & pointers,	https://www.cs.fsu.edu/~myers/cop3330/notes/dma.html	https://www.youtube.com/watch?v=vIcOhM_Vkc4
60-63	Program structure of C++. Conversion of Algorithm into Program.	https://home.csulb.edu/~pnguyen/cecs100/lecturenotes/Programming%20Developmen	https://www.youtube.com/watch?v=VEB8BcI5u10
64-67	Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting from Linked List	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1188/lectures/Lecture16/Lecture16.pdf	https://www.youtube.com/watch?v=6wXZ_m3SbEs
68-70	Operations on Doubly Linked List, Circular linked List & its applications	https://web.eecs.utk.edu/~bvanderz/teaching/cs140Fa10/notes/Dllists/	https://www.youtube.com/watch?v=xXNdPe17dtg
71-73	Stacks: Array representation of Stacks, implementation of stack using linked list.	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1186/lectures/05-	https://www.youtube.com/watch?v=08QSylWv6jM
74-81	Applications: Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1186/lectures/05-Stacks_Queues/5-Stacks_Queues.pdf	https://www.youtube.com/watch?v=XkLfiA7Xbks
82-86	Queues: Representation as Array and Linked List.	https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1186/lectures/05-Stacks_Queues/5-Stacks_Queues.pdf	https://www.youtube.com/watch?v=XkLfiA7Xbks
87-90	Deques, Circular Queues, Priority Queues.	https://web.eecs.utk.edu/~bvanderz/teaching/cs140Fa10/notes/Queues/	https://www.youtube.com/watch?v=2zQtymZV6dk
91-98	Trees: Binary trees, complete binary trees, Binary Search Trees, Data structures for representing binary trees, Insertion, Deletion and Searching of Binary Search	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec6/	https://ocw.mit.edu/courses/6-851-advanced-data-structures-spring-2012/resources/session-15-static-trees/

99-107	Tree and their Implementation, Tree Traversal: preorder, In order, Post order and their algorithms, Their Implementation. B Trees.	http://webdocs.cs.ualberta.ca/~holte/T26/tree-traversal.html	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-6-binary-trees-part-1/
108-117	Heaps, Difference between heap and Array, insertion and deletion in heap, Heap sort implementation and its applications.	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec8/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-8-binary-heaps/
118-120	Graphs: Basic terminology, directed and undirected graphs, notion of path.	https://www.seas.upenn.edu/~jean/cis160/cis260slides5.pdf	https://www.youtube.com/watch?v=LxN4oUWJNag
121-125	Representation of graphs: edge list structures, adjacency list structures, adjacency matrix, Linked List representation of Graph.	https://web2.qatar.cmu.edu/~mhhammou/15122-s16/lectures/23-graphs.pdf	https://www.youtube.com/watch?v=JONnqF-oCDo
126-128	Operations on Graph, Graph traversals: DFS, BFS and their implementation.	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec9/ https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec10/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-9-breadth-first-search/ https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-10-depth-first-search/
129-131	Hashmap, Hashing Techniques, Collision and its resolving. Trie Data Structure, Basic Operations on Trie.	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec4/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-4-hashing/
132-136	Introduction to Dynamic Programming, Standard problems on Dynamic Programming, One Dimensional Dynamic Programming.	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec15/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-15-dynamic-programming-part-1-srtbot-fib-dags-bowling/
137-140	Two Dimensional Dynamic Programming, Top DP Algorithms Greedy Algorithms	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/mit6_006s20_lec16/	https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/resources/lecture-16-dynamic-programming-part-2-lcs-lis-coins/

9. Action plan for different types of learners

Slow Learners	Average Learners	Fast Learners
• Remedial Classes on Saturdays • Offer supplementary materials or activities to reinforce concepts outside regular class hours. • Leverage educational apps, interactive software, and online resources tailored to individual needs.	• Incorporate hands-on activities, visual aids, and interactive discussions to keep students engaged. • Encourage students to share their strengths and support each other in areas of challenge.	• Incorporate real-life problem-solving scenarios to engage fast learners in critical thinking. • Encourage the pursuit of independent research projects or special assignments aligned with personal interests. • Offer optional extension activities for those who wish to explore topics in greater depth.

10. Evaluation Scheme & Components:

Evaluation Component	Type of Component	No. of Assessments	Weightage of Component	Mode of Assessment
Component 2	Subjective Test/Sessional Tests (STs)	03*	40%	Offline
Component 3	End Term Examinations	01	60%	Offline
Total		**100%**		

***All the STs are mandatory. Makeup Examination will compensate for either ST1 or ST-2 (Only for genuine cases, based on Dean's approval).**

11. Details of Evaluation Components:

Evaluation Component	Description	Syllabus Covered (%)	Timeline of Examination	Weightage (%)
Component 2	ST 01	32%	Week 6	40%
	ST 02	33% - 64%	Week 10	
	ST 03	65% - 93%	Week 17	
Component 3	End Term Examination*	100%	At the end of the semester	60%
Total				100%

*As per Academic Guidelines minimum 85% attendance is required to become eligible for appearing in the End Semester Examination.

12. Syllabus of the Course:

Lecture Number	Topics	No. of Lectures	Weightage %
1-9	Data Structures and Algorithms: Basic Terminology, Elementary Data Organization, Data Structures and Operations, Algorithm : Complexity, Time and Space & Complexity, Asymptotic Notations for Complexity(Ω, ω ,θ, O, o)	**9**	**32%**
10-27	Array: Introduction, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Insertion and Deletion in arrays. Searching: Linear and Binary Search with their Complexity. Character Arrays, Strings, Declaration and Initialization of character array, Memory Representation, Basic Operations. Sorting a Char Array. Sorting techniques: Selection Sort, Insertion Sort, Quick Sort, Merge Sort.	**18**	
28-45	Bitmasking, Bitwise operations in bitmasking, STL (Standard Template Library), Components of the STL. Recursion & Backtracking: Introduction Recursion and Recursive Function, Deep Diving into recursion, Implementation of Recursion on Subsets, Backtracking Algorithms,	**18**	
ST-1 (Syllabus = Lecture number 1-45)			
46-63	Preliminaries of Object oriented programming using C++: keywords, identifiers, data types, Decision control and looping statements, Functions, Basics of objects and classes, Dynamic Memory Management & pointers, Program structure of C++. Conversion of Algorithm into Program.	**18**	**32%**
64-70	Linked List: Introduction & its memory representation, traversing a Linked List, Insertion into Linked List (sorted and unsorted Linked List), Deleting from Linked List. Operations on Doubly Linked List, Circular linked List & its applications	**7**	
71-90	Stacks: Array representation of Stacks, implementation of stack using linked list, Applications: Arithmetic Expressions, Polish Notation, Transforming Infix Expressions into Postfix Expressions, Implementations of recursive and non-recursive procedures by Stacks Queues: Representation as Array and Linked List, Deques, Circular Queues, Priority Queues.	**20**	
ST-1 (Syllabus = Lecture number 46-90)			
91-117	Trees: Binary trees, complete binary trees, Binary Search Trees, Data structures for representing binary trees, Insertion, Deletion and Searching of Binary Search Tree and their Implementation, Tree Traversal: preorder, In order, Post order and their algorithms, Their Implementation. Heaps, Difference between heap and Array, insertion and deletion in heap, Heap sort implementation and its applications. B Trees.	**27**	**29%**

118-131	Graphs: Basic terminology, directed and undirected graphs, notion of path. Representation of graphs: edge list structures, adjacency list structures, adjacency matrix, Linked List representation of Graph, Operations on Graph, Graph traversals: DFS, BFS and their implementation. Hashmap, Hashing Techniques, Collision and its resolving. Trie Data Structure, Basic Operations on Trie.	**14**	
ST-3 (Syllabus = Lecture number 91-131)			
132-140	Introduction to Dynamic Programming, Standard problems on Dynamic Programming, One Dimensional Dynamic Programming, Two Dimensional Dynamic Programming, Top DP Algorithms Greedy Algorithms	**9**	**7%**
End Term (Covering 100% syllabus)			

This Document is approved by:

Designation	**Name**	**Signature**
Course Coordinator	Dr. Anuj Kumar Jain	
Head-Academic Delivery	Dr. Ravi Kumar Sachdeva	
Dean	Dr. Raj Gaurang Tiwari	
Dean Academics	Dr. Monit Kapoor	
Date (DD/MM/YYYY)	Dec 06, 2023	