Name of Institute: IITE Indus University Ahmedabad
Name of Faculty: Prof. Khushbu Maurya

Course code: CE0417
Course name: Data Structure and Algorithm
Pre-requisites: Knowledge of c/c++/java, Flow chart
Credit points: 4
Offered Semester: 4th

Course Coordinator (weeks 12)
Full Name: Prof Hiren Mer
Department with sitting location: 4th floor Bhanvar Building
Telephone:9227885688
Email: hirenmer.ce@indusuni.ac.in
Consultation times: 3.00 P.M. to 5.00 P.M. (Monday to Friday)

Course Lecturer (weeks 12)
Full Name:
Prof Hiren Mer, Dr. Nimisha Patel, Prof. Khushbu Maurya, Prof. Zalak Trivedi ,Prof. Dhaval Patel
Department with sitting location:4th floor Bhanvar Building
Email: hirenmer.ce@indusuni.ac.in, nimishapatel.ce@indusuni.ac.in , khushbumaurya.ce@indusuni.ac.in , zalaktrivedi.ce@indusuni.ac.in , dhavalpatel.ce@indusuni.ac.in
Consultation times: **3**.00 P.M. to 5.00 P.M. (Monday to Friday)

Students will be contacted throughout the Session via Mail with important information relating to this Course.

Course Objectives

- To enable an efficient storage of data for an easy access.
- To enable an efficient processing of data.
- To help in data protection and management.

Course Outcomes (CO)

After successful completion of the course, student will able:

1. Explain how arrays, stacks, queues, linked lists, trees, heaps, Graphs and Hash Tables are represented in the main memory and manipulated or used by different operations. [BT 4]
2. Construct algorithms for performing operations on a data structure, with an understanding of the trade-off between the time and space complexity. [BT 6]
3. Compare alternate implementations of an Abstract Data Type with respect to their performance. [BT 5]
4. Illustrate how arrays, stacks, queues, linked lists, trees, heaps, Graphs and Hash Tables are used in various applications. [BT 4]
5. Analyse the computational efficiency of key searching, sorting and Hashing algorithms.[BT 4]

6. Evaluate the suitability of different data structures for solving computing problems.[BT 5]

Course Outline

In this course, we consider the common data structures that are used in various computational problems. You will learn how these data structures are implemented in different programming languages and will practice implementing them in our programming assignments. This will help you to understand what is going on inside a particular built-in implementation of a data structure and what to expect from it.

Method of delivery

(Face to face lectures, self-study material, Active Learning Techniques)

Study time

3 hrs (Theory) + 2 hrs (Practicals)

CO-PO Mapping (PO: Program Outcomes)

1 Program Outcomes (PO's)

Engineering Graduates will be able to:

- **PO1** **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- **PO2** **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- **PO3** **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- **PO4** **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **PO5** **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
- **PO6** **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- **PO7** **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- **PO8** **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- **PO9** **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

2. Programme Specific Outcome

Computer Engineering

1. To understand the principles and working of computer systems.
2. To Design and develop computer programs in the areas related to algorithms, networking, web design, cloud computing, IoT and data analytics.
3. Should able to understand the structure and development methodologies of software systems with the use of a various programming languages and open source platforms
4.

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO110	PO111	PO112	PSO1	PSO2
1	3	1	1	1	-	-	-	-	-	-	-	-	3	3
2	3	1	3	2	-	-	-	-	-	-	-	-	2	1
3	1	3	2	-	-	-	-	-	-	-	-	-	1	1
4	2	1	2	-	-	-	-	-	-	-	-	-	2	1
5	3	3	2	1	2	1	-	-	-	-	-	-	1	2
6	3	3	2	1	-	-	-	-	2	-	-	-	1	1

Blooms Taxonomy and Knowledge retention (For reference)

(Blooms taxonomy has been given for reference)

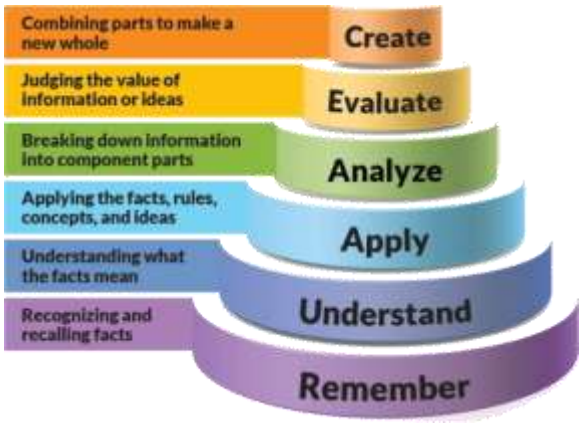

Figure 1: Blooms Taxonomy

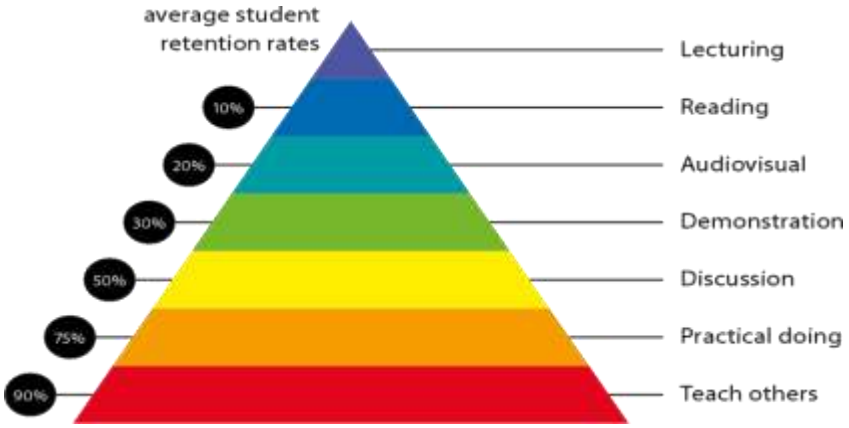

Figure 2: Knowledge retention

Practical work:

Wk No.	Class Activity	List of Practical
01	Lab 1	Write a program to implement following searching algorithms. Linear (2) Binary.
02	Lab 2	Write a program to implement following sorting algorithms. Bubble (2) Selection (3) Quick (4) Merge
03	Lab 3	Write a program to implement following STACK operations. PUSH (2) POP (3) PEEP (4) CHANGE (5) DISPLAY
04	Lab 4	Write a program to convert infix expression to postfix expression.
05	Lab 5	Write a program to implement following QUEUE operations. INSERT (2) DELETE (3) DISPLAY
06	Lab 6	Write a program to implement following CQUEUE operations. INSERT (2) DELETE (3) DISPLAY
07	Lab 7	Write a program to implement following operations of the singly linked list (SLL). (1) Insert a node at the front of the linked list. (2) Insert a node at the end of the linked list. (3) Insert a node such that linked list is in ascending order. (According to info. Field)
08	Lab 8	Write a program to implement following operations of the singly linked list (SLL). (1) Delete a first node of the linked list. (1) Delete a node before specified position. Delete a node after specified position.
09	Lab 9	Write a program to implement following operations of the doubly linked list (DLL). (1) Insert a node at the front of the linked list. Insert a node at the end of the linked list.
10	Lab 10	Write a program to implement following operations of the doubly linked list (DLL). (1) Delete a last node of the linked list. Delete a node before specified position.
11	Lab 11	Write a program to implement stack using linked list.
12	Lab 12	Write a program to implement queue using linked list.

13	Lab 13	Write a program to implement binary tree traversals.
Practical Beyond syllabus		
14	Lab 14	Write a Program to implement heap sort.
15	Lab 15	Write a Program to implement priority queue.

Lecture/tutorial times

(Give lecture times in the format below)

CE-A

Lecture	Tuesday	11:10 to 12:10 p.m.
Lecture	Wednesday	11:10 to 12:10 p.m.
Lecture	Friday	2.00 to 3.00 p.m.

Attendance Requirements

The University norms states that it is the responsibility of students to attend all lectures, tutorials, seminars and practical work as stipulated in the Course outline. Minimum attendance requirement as per university norms is compulsory for being eligible for mid and end semester examinations.

Details of referencing system to be used in written work

Text books:

Text Books

1. An Introduction to data structures with applications. By jean paul tremblay & paul G Sorenson publisher TMH.

Reference Books

1. Data Structures using C & C++ -By Ten Baum Publisher – Prenctice-Hall International.
2. Fundamentals of Data Structures in C++-By Sartaj Sahani.
3. Classical Data Structure by D. samantha. Pearson publication.

Additional Materials

https://www.studytonight.com/data-structures/introduction-to-data-structures
https://www.javatpoint.com/data-structure-tutorial
https://www.tutorialspoint.com/data_structures_algorithms/data_structures_basics.htm

ASSESSMENT GUIDELINES

Your final course mark will be calculated from the following:

Theory:	*Practical:*
Class Test [40 Marks]	Practical performance [20 Marks]
	Internal Exam + Viva/
Assignment [10 Marks]	Practical Mini project [20 Marks]
attendance bonus for all students having	Regularity in Lab+ Practical
attendance > 80% [05 Marks]	Manual+ Viva [20 marks]
presentation [05 Marks]	

SUPPLEMENTARY ASSESSMENT

Students who receive an overall mark less than 40% in mid semester or end semester will be considered for supplementary assessment in the respective components (i.e mid semester or end semester) of semester concerned. Students must make themselves available during the supplementary examination period to take up the respective components (mid semester or end semester) and need to obtain the required minimum 40% marks to clear the concerned components.

Practical Work Report/Laboratory Report:

A report on the practical work is due the subsequent week after completion of the class by each group.

Late Work

Late assignments will not be accepted without supporting documentation. Late submission of the reports will result in a deduction of -% of the maximum mark per calendar day

Format

All assignments must be presented in a neat, legible format with all information sources correctly referenced. **Assignment material handed in throughout the session that is not neat and legible will not be marked and will be returned to the student.**

Retention of Written Work

Written assessment work will be retained by the Course coordinator/lecturer for two weeks after marking to be collected by the students.

University and Faculty Policies

Students should make themselves aware of the University and/or Faculty Policies regarding plagiarism, special consideration, supplementary examinations and other educational issues and student matters.

Plagiarism - Plagiarism is not acceptable and may result in the imposition of severe penalties. Plagiarism is the use of another person's work, or idea, as if it is his or her own - if you have any doubts

at all on what constitutes plagiarism, please consult your Course coordinator or lecturer. Plagiarism will be penalized severely.

Do not copy the work of other students.
Do not share your work with other students (except where required for a group activity or assessment)

Course schedule (subject to change)

(Mention quiz, assignment submission, breaks etc as well in the table under the Teaching Learning Activity Column)

	Week #	Topic & contents	CO Addressed	Teaching Learning Activity (TLA)
	Weeks 1	**INTRODUCTION TO DATA STRUCTURE :** Definition, classification of data structure, Examples of data structure.	1,2	Chalk & BB/Online Session with PPT
	Weeks 2	**Searching and Sorting:** Various sorting techniques: Selection sort - bubble sort - Quick sort, Merge sorting. Sequential searching, Binary searching	1,2,5	Chalk & BB/Online Session with PPT
	Week 3	**LINEAR DATA STRUCTURE:** Representation of arrays, Applications of arrays, Stack: Stack-Definitions & Concepts, Operations On Stacks, Applications of Stacks	*1,2,3*	Chalk & BB/Online Session with PPT
	Week 4	**LINEAR DATA STRUCTURE:** Polish Expression, Reverse Polish Expression And Their Compilation, Recursion, Tower of Hanoi.	1,2,3	Chalk & BB/Online Session with PPT
	Week 5	**Queue:** Representation Of Queue, Operations On Queue, Circular Queue, Priority Queue, Double Ended Queue, Applications of Queue	1,2	Chalk & BB/Online Session with PPT
	Week 6	**Linked List** Singly Linked List, Doubly Linked list	*2,3*	Chalk & BB/Online Session with PPT
	Week 7	Circular linked list, Linked implementation of Stack, Linked implementation of Queue, Applications of linked list.	1,2	Chalk & BB/Online Session with PPT
	Week 8	**NONLINEAR DATA STRUCTURE:** Tree-Definitions and Concepts, Representation of binary tree, Binary tree traversal (In order, post order, pre	2,3	Chalk & BB/Online Session with PPT

		order), Threaded binary tree, Binary search trees,		
	Week 9	**Applications of Tree:** Some balanced tree mechanism, eg. AVL trees, 2-3 trees, Height Balanced, Weight Balance, Graph-Matrix Representation Of Graphs,	*1,2*	Chalk & BB/Online Session with PPT
	Week 10	Elementary Graph operations(Breadth First Search, Depth First Search, Spanning Trees, Shortest path, Minimal spanning tree)	1,2,3	Chalk & BB/Online Session with PPT
	Week 11	**HASHING :** Hashing: The symbol table, Hashing Functions, Collision-Resolution Techniques	1,2,5	Chalk & BB/Online Session with PPT
	Week 12	Indexing structure for index files, hashing for direct files, Multi-Key file organization and access methods	1,2,3	Chalk & BB/Online Session with PPT

PROGRAM MAP for Bachelor of Engineering
(CE / CSE / IT)