GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)
Semester – III

Course Title: Data Structures and Algorithms
(Course Code: 4330704)

Diploma Programme in which this course is offered	Semester in which offered
Computer Engineering	Third

1. RATIONALE

Development of application systems and software that use underlying architecture of machines efficiently and effectively requires the ability to use and manipulate various types of Data Structures and other constructs. This being a fundamental ability which is language neutral, yet requires use of a language for its implementation. This is a basic course which goes along with other programming courses to develop an integrated ability to efficient software development, hence this course is very important for computer engineers.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Implement various types of algorithms using Data Structures.**

3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

The practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency:

a. Perform basic operations on arrays and strings.
b. Demonstrate algorithms to insert and delete elements from the stack and queue data structure.
c. Apply basic operations on the linked list data structure.
d. Illustrate algorithms to insert, delete and searching a node in tree.
e. Apply different sorting and searching algorithms to the small data sets.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (CI+T/2+P/2)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	CA	ESE	CA	ESE	
3	0	4	5	30	70	25	25	150

Out of 30 marks under the theory CA, 10 marks are for assessment of the micro-project to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessing the attainment of the cognitive domain UOs required for the attainment of the COs.

Legends: CI-ClassRoom Instructions; T – Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, CA - Continuous Assessment; ESE - End Semester Examination.

5. SUGGESTED PRACTICAL EXERCISES

The following practical outcomes (PrOs) that are the sub-components of the COs. *Some of the **PrOs** marked **'*'** are compulsory, as they are crucial for that particular CO. These PrOs need to be attained at least at the 'Precision Level' of Dave's Taxonomy related to 'Psychomotor Domain'*.

S.No.	Practical Outcomes (PrOs)	Uni t No.	Approx. Hrs. Required
1	Define various terms such as algorithm, various approaches to design an algorithm, time complexity, space complexity, best case, average case and worst-case time complexity etc.	I	02
2	Implement array using row major order and column major order	I	02
3	Implement Sequential search algorithms.	I	02
4	Implement Binary search algorithms.	I	02
5	Implement various string algorithms.	II	02
6	Implement push and pop algorithms of stack using array	III	02
7	Implement recursive functions.	III	02
8	Implement insert algorithms of queue using array.	III	02
9	Implement delete algorithms of queue using array.	III	02
10	Implement simple structure programs using pointers.	IV	02
11	Implement insertion of node in the beginning of the list in singly linked list.	IV	02
12	Implement insertion of node at the end of list in singly linked list.	IV	02
13	Implement insertion of node in sorted linked list.	IV	02
14	Implement insertion of node at any position in liked list.	IV	02

15	Implement counting no of node algorithm in singly linked list.	IV	02
16	Implement searching of a node algorithm in singly linked list.	IV	02
17	Implement delete a node algorithm in singly linked list.	IV	02
18	Implement construction of binary search tree.	V	02
19	Implement inorder, preorder and postorder traversal methods in binary search tree.	V	04
20	Implement searching algorithm in binary search tree.	V	02
21	Implement Bubble sort algorithm.	VI	02
22	Implement Selection sort algorithm.	VI	02
23	Implement Quick Sort algorithm.	VI	02
24	Implement Insertion sort algorithm.	VI	02
25	Implement Shell sort algorithm.	VI	02
26	Implement Merge Sort algorithm.	VI	02
27	Solve hash table example using division method, method square method, folding method. (paper work only)	VI	02
	Total		56

Note

i. *More **Practical Exercises** can be designed and offered by the respective course teacher to develop the industry relevant skills/outcomes to match the COs. The above table is only a suggestive list.*

*ii. The following are some **sample** 'Process' and 'Product' related skills (more may be added/deleted depending on the course) that occur in the above listed **Practical Exercises** of this course required which are embedded in the COs and ultimately the competency.*

S. No.	Sample Performance Indicators for the PrOs	Weightage in %
1	Correctness of program	30
2	Readability and documentation of the program/Quality of input and output displayed (messaging and formatting)	10
3	Code efficiency	20
4	Debugging ability	20
5	Program execution/answer to sample questions	20
Total		**100**

6. MAJOR EQUIPMENT/ INSTRUMENTS AND SOFTWARE REQUIRED

These major equipment/instruments and Software required to develop PrOs are given below with broad specifications to facilitate procurement of them by the administrators/management of the institutes. This will ensure conduction of practical in all

institutions across the state in proper way so that the desired skills are developed in students.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Computer with latest configuration with windows or unix os	All
2	C/C++/Python Compiler	All

7. AFFECTIVE DOMAIN OUTCOMES

The following ***sample*** Affective Domain Outcomes (ADOs) are embedded in many of the above-mentioned COs and PrOs. More could be added to fulfil the development of this competency.

a) Follow safety practices.
b) Practice good housekeeping.
c) Demonstrate working as a leader/a team member.
d) Maintain tools and equipment
e) Follow ethical practices.

The ADOs are best developed through the laboratory/field-based exercises. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

i. 'Valuing Level' in 1st year
ii. 'Organization Level' in 2nd year.
iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major Underpinning Theory is formulated as given below and only higher level UOs of *Revised Bloom's taxonomy* are mentioned for development of the COs and competency in the students by the teachers. (Higher level UOs automatically includes lower level UOs in them). If required, more such higher level UOs could be included by the course teacher to focus on attainment of COs and competency.

Unit	Unit Outcomes (UOs)	Topics and Sub-topics
Unit – I Basic Concepts of Data Structures	1a. Represent the data in relevant memory	1.1 Data Structure Basic Concepts 1.2 Types of data structures
	1b. Differentiate primitive and non-primitive data structures	1.3 Primitive and non-primitive data structures
	1c. List key features of an algorithm	1.4 Introduction to Algorithms 1.5 Key features of an algorithm
	1d. Define time complexity and space complexity	1.6 Analysis Terms (for the definitions purpose only): a. Time Complexity b. Space Complexity c. Asymptotic Notations: Big Oh Notation, Big Omega Notation, Big Theta Notation, Best case Time Complexity, Average case Time Complexity, Worst case Time Complexity
	1e. Implement programs to represent array in row major and column major order	1.7 Array: i. Row Major Arrays ii. Column Major Arrays 1.8 Overview of different array operations.
	1f. Design and Implement search algorithms	1.9 Searching an element into an array: i. Linear Search ii. Binary Search
Unit– II Strings	2a. Describe representation of a strings	2.1 String representation: Reading and Writing Strings
	2b. Develop algorithms to implement various operations on string	2.2 String operations : Finding length of a string, Converting Characters of a string into upper case and lower case, Concatenation of two strings to form a new string, Appending, Reversing a string, Copying a string, Comparing strings, Insertion, Substring, Deletion

Unit– III Stack and Queues	3a. Define linear and non- linear data structures 3b. Implement algorithms to push an element into stack, pop an element from the stack.	3.1 Linear and Non-Linear Data Structures 3.2 Stack : Array representation of Stack, PUSH- POP Operations on Stack, Implementation of Stack, Application of Stack, Infix, Prefix and Postfix Forms of Expressions, Recursive Functions (Factorial, greatest common divisor, Fibonacci series)
	3c.. Implement different operations on a Queue	3.3 Queue: Array representation of Queue Operations on a Queue (Add an element, delete an element, display all elements of a queue), Implementation of a Queue, Limitation of a Single Queue
	3c. Differentiate circular and simple queue	3.4 Concepts of Circular Queue 3.5 Applications of a queue 3.6 Differentiate circular queue and simple queue
Unit– IV Linked List	4a. Define linked list	4.1 Pointers Revision 4.2 Revision of Structure 4.3 Revision of structure using pointers 4.4 Dynamic Memory Allocation 4.5 Linked list Presentation 4.6 Types of Linked List
	4b. Implement algorithms to perform various operations on a singly linked list	4.7 Basic operations on singly linked list : Insertion of a new node in the beginning of the list, at the end of the list, after a given node, before a given node, in sorted linked list, Deleting the first and last node from a linked list, Searching a node in Linked List, Count the number of nodes in linked list
	4c. Distinguish circular linked list and singly linked list	4.8 Concepts of circular linked list 4.9 Difference between circular linked list and singly linked list
	4d. Distinguish Doubly linked list and singly linked list	4.10 Doubly linked list: Representation 4.11 Difference between Doubly linked list and singly linked list
	4e. List applications of the linked list	4.12 Applications of the linked list
Unit– V Trees	5a. Define non-linear data structure	5.1 Non-linear data structures: Tree, Graph
	5b. Define basic terms of a tree data structure 5c. Convert general tree to binary tree	5.2 Basic Terms: General Tree, Forest, Binary trees, level number, degree, in-degree and out-degree, root node, leaf node, directed edge, path, depth Binary tree: Complete Binary Tree, Strict Binary Tree, Conversion of General Tree to Binary Tree

	5d. Implement basic operations on a binary tree 5e. Demonstrate the traversal of a binary tree	5.3 Binary Search Tree : Insertion of a node in binary tree, Deletion of a node in binary tree, Searching a node in binary tree 5.4 Binary Tree Traversal : Inorder, Preorder, Postorder
	5f. List applications of tree	5.5 Applications of binary tree
Unit– VI Sorting and Hashing	6a. Arrange data in ascending and descending orders using appropriate sorting algorithm	6.1 Sorting Methods : a. Bubble Sort, b. Selection Sort, c. Quick Sort, d. Insertion Sort, e. Merge Sort, f. Radix Sort
	6b. Apply various hashing techniques	6.2 Hashing Concepts 6.3 Hash functions: Division Method, Middle Square Method, Folding Method

***Note**: The UOs need to be formulated at the 'Application Level' and above of Revised Bloom's Taxonomy' to accelerate the attainment of the COs and the competency.*

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Basic Concepts of Data Structures	4	4	3	0	7
II	Strings	4	2	2	3	7
III	Stack and Queues	8	2	6	6	14
IV	Linked List	9	4	8	2	14
V	Trees	7	4	4	6	14
VI	Sorting and Hashing	10	2	6	6	14
Total		**42**	**18**	**29**	**23**	**70**

***Legends:** R=Remember, U=Understand, A=Apply and above (Revised Bloom's taxonomy)*
***Note**: This specification table provides general guidelines to assist student for their learning and to teachers to teach and question paper designers/setters to formulate test items/questions assess the attainment of the UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.*

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related

co-curricular activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare small reports (of 1 to 5 pages for each activity). For micro project report should be as per suggested format, for other activities students and teachers together can decide the format of the report. Students should also collect/record physical evidences such as photographs/videos of the activities for their (student's) portfolio which will be useful for their placement interviews:

a) Students are encouraged to learn Visual Language programming like scratch, snap etc.
b) Undertake micro-projects in teams.
c) Prepare charts to explain use/process of the identified topic.
d) https://www.codechef.com/ , in this website very elementary programs are available, students are expected to solve those programs
e) https://code.org/, an hour of code may be organized and students are encouraged to participate
f) Students are encouraged to register themselves in various MOOCs such as: Swayam, edx, Coursera, Udemy etc to further enhance their learning.
g) List the applications which are developed using C
h) Encourage students to participate in different coding competitions like hackathon, online competitions on codechef etc.
i) Encourage students to form a coding club at institute level and can help the slow learners

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a) Massive open online courses (***MOOCs***) may be used to teach various topics/sub topics.
b) Guide student(s) in undertaking micro-projects.
c) Managing Learning Environment
d) Diagnosing Essential Missed Learning concepts that will help for students.
e) Guide Students to do Personalized learning so that students can understand the course material at his or her pace.
f) Encourage students to do Group learning by sharing so that teaching can easily be enhanced.
g) ***'CI" in section No. 4***means different types of teaching methods that are to be employed by teachers to develop the outcomes.
h) About ***20% of the topics/sub-topics*** which are relatively simpler or descriptive in nature is to be given to the students for ***self-learning***, but to be assessed using different assessment methods.
i) With respect to ***section No.11***, teachers need to ensure to create opportunities and provisions for ***co-curricular activities***.
j) Guide students on how to address issues on environment and sustainability using the knowledge of this course

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based (group of 3 to 5). However, **in the fifth and sixth semesters**, the number of students in the group should ***not exceed three.***

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total work load on each student due to the micro-project should be about ***16 (sixteen) student engagement hours*** (i.e., about one hour per week) during the course. The students ought to submit micro-project by the end of the semester (so that they develop the industry-oriented COs).

A suggestive list of micro-projects is given here. This should relate highly with competency of the course and the COs. Similar micro-projects could be added by the concerned course teacher:

a) Develop a C/C++/Python Program that evaluate the given arithmetic expression using stack.
b) Develop a C/C++/Python Program that maintain a queue of persons. In this queue user can add a person, delete a person and search a person.
c) Develop a C/C++/Python Program that perform banking operations like withdraw cash, deposit cash and mini statement using appropriate data structure.
d) Develop a C/C++/Python Program for process management algorithm by using appropriate data structure.
e) Develop a C/C++/Python Program for print spooler using appropriate Data structure.
f) Develop a C/C++/Python Program for Telephone Directory system. In this user can adding, searching, modifying, listing, and deleting records through the use of appropriate data structure.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	An Introduction to Data Structures with Applications	Jean-Paul Tremblay & Paul G. Sorenson	Tata McGraw Hill
2	Data and File Structures using C	Thareja,Reema	Oxford University Press New
3	Data Structures	Chitra, A Rajan, P T	Tata McGraw Hill, New delhi,
4	Data Structures using C & C++	Tenen Baum	Prenctice-Hall International
5	Classic Data Structures	Samanta, D.	PHI Learning, New Delhi
6	Data Structures using C	ISRD Group	McGraw Hill, New Delhi
7	Data Structures: A Pseudo-code approach with C	Gilberg & Forouzan	Thomson Learning

14. SUGGESTED LEARNING WEBSITES

a) https://www.programiz.com/dsa
b) https://nptel.ac.in/courses/106102064 (Introduction to data structures and algorithms, IIT Delhi)
c) https://nptel.ac.in/courses/106106133 (Programming, Data structures and Algorithms, IIT Madras)
d) https://www.codecademy.com/learn/linear-data-structures
e) https://www.udacity.com/course/data-structures-and-algorithms-in-python--ud513
f) https://www.edx.org/learn/data-structures

15. PO-COMPETENCY-CO MAPPING

Semester III	Data Structures and Algorithms (Course Code:4330704)									
	POs and PSOs									
Competency & Course Outcomes	**PO 1 Basic & Discipline specific knowledge**	**PO 2 Problem Analysis**	**PO 3 Design/development of solutions**	**PO 4 Engineering Tools, Experimentation &Testing**	**PO 5 Engineering practices for society, sustainability & environment**	**PO 6 Project Management**	**PO 7 Life-long learning**	**PSO 1**	**PSO 2**	**PSO 3** (If need e d)
Competency Implement various types of algorithms using Data Structures										
Course Outcomes a) Perform basic operations on arrays and strings	**2**	**1**	**2**	**-**	**-**	**-**	**-**			
b) Demonstrate algorithms to insert and delete elements from the stack and queue data structure	**3**	**1**	**2**	**-**	**-**	**-**	**-**			
c) Apply basic operations on the linked list data structure.	**3**	**-**	**2**	**-**	**-**	**-**	**-**			
d) Illustrate algorithms to insert, delete and searching a node in tree.	**2**	**1**	**1**	**-**	**-**	**-**	**-**			
e) Apply different sorting and searching algorithms to the small data sets.	**3**	**2**	**2**	**-**	**-**	**-**	**-**			

Legend: '**3**' for high, '**2**' for medium, '**1**' for low or '**-**' for the relevant correlation of each competency, CO, with PO/ PSO

16. COURSE CURRICULUM DEVELOPMENT COMMITTEE GTU

Resource Persons

S. No.	Name and Designation	Institute	Contact No.	Email
1	Shri P. P. Kotak Principal	Government Polytechnic Rajkot	8200601748	kotakp2003@yahoo.com
2	Smt. M. P. Mehta Head of the Department	Government Polytechnic, Gandhinagar	9879578273	manishamehtain@gma il.com
3	Smt. Rashmika K. Vaghela Lecturer	K.D.Polytechnic, Patan	9824464199	rashmivaghela.rv@gmail.com
4	Ms. Kumundrini B Prajapati	Government Polytechnic, Gandhinagar	9974543026	kumundrini13187@gmail.com