St. MARTIN's ENGINERING COLLEGE

Dhulapally,Secunderabad-500014

INFORMATION TECHNOLOGY
COURSE DESCRIPTION FORM

Course Title	Data Structures			
Course Code	A30502			
Regulation	R13-JNTUH			
Course Structure	Lectures	Tutorials	Practicals	Credits
	4	-	3	4
Course Coordinator	Mrs. E.Soumya, Assistant Professor, IT			
Team of Instructors	Ms. A.Santhoshi, Assistant Professor, IT Mr.B.Subbarayudu, Assistant Professor, IT			

I.Course Overview:

A data structure is a subject of primary importance to the discipline of information technology. It is a logical and mathematical model of sorting and organizing data in a particular way in a computer, required for designing and implementing efficient algorithms and program development. Different kinds of data structures like arrays, linked lists, stacks, queues etc. are suited to different kinds of applications. Some specific data structures are essential ingredients of many efficient algorithms, make possible the management of huge amounts of data, such as large databases and internet indexing services. Now a day's various programming languages like C,C++ and java are used to implement the concepts of data structures, of which C remains the language of choice for programmers across the world.

II. Prerequisite(s)

Level	Credits	Periods/Week	Prerequisites
UG	4	6	**Basic data structures, discrete mathematics**

III. Marks Distribution:

Sessional Marks	University EndExam Marks	Total Marks
There shall be two midterm examinations. Each midterm examination consists of essay paper, objective paper and assignment.		

The essay paper is for 10 marks of 60 minutes duration and shall contain 4 questions. The student has to answer 2 questions, each carrying 5 marks. The objective paper is for 10 marks of 20 minutes duration. It consists of 10 multiple choice and 10 fill in the blank questions and each carries half mark. First midterm examination shall be conducted for the first two and half units of syllabus and second midterm examination shall be conducted for the remaining portion. Five marks are earmarked for assignments. There shall be two assignments in every theory course. Assignments are usually issued at the time of commencement of the semester. These are of problem solving in nature with critical thinking. Marks shall be awarded considering the average of two midterm tests in each course.	75	100

IV. Evaluation Scheme:

S.No	Component	Duration	Marks
1	I Mid Examination	80 minutes	20
2	I Assignment	-	5
3	II Mid Examination	80 minutes	20
4	II Assignment	-	5
5	External Examination	3 hours	75

V. Course Objectives:

At the end of the course, the students will be able to:

1. Understand the basic concepts such as abstract data types, Linear and non linear data structures.

2. Describe the notations used to analyze the performance of algorithm.

3. Demonstrate the behavior of data structures such as stacks, queues, trees, hash tables, search trees, graphs and their representations.

4. Choose the appropriate data structures for a specified application.

5. Discuss and analyze various searching and sorting algorithms.

VI. Course Outcomes:

After completing this course the student must demonstrate the knowledge and ability to:

1. **Understand** various recursive methods**.**
2. **Compare** iterative and recursive solutions for elementary problems.
3. **Analyze** various algorithms and its space and time complexity.
4. **Solve** problems using various data structures like linear list, stack, queue, tree and graphs.
5. **Design** and apply appropriate tree and graph data structure for solving computing problems.
6. **Analyze** various searching and sorting techniques.
7. **Analyze** the associated algorithms, operations and time complexity.
8. **Understand** the various search trees and their complexities.

VII. How program outcomes are assessed:

Program outcomes		**Level**	**Proficiency assessed by**
PO1	Ability to apply acquired knowledge of science and engineering fundamentals in problem solving.	**H**	Assignments, Exercises
PO2	Ability to undertake problem identification, formulation and providing optimum solution in software applications.	**H**	Exercises
PO3	Ability to utilize systems approach in designing and to evaluate operational of developed software.	**H**	Exercises
PO4	Graduate will be capable to use modern tools and packages available for their professional area.	**S**	--
PO5	Ability to identify, formulate and solve complex information technology related problems.	**N**	--
PO6	Understanding of the social, cultural responsibilities as a professional engineering in a global context.	**N**	--
PO7	Ability to understand the impact of environment on engineering designs based on the principles of inter disciplinary domains for suitable development.	**N**	--
PO8	Ability to understand the role of ethics in professional environment and implementing them.	**S**	Seminars
PO9	Competency in software development to function as an individual and in a team of multidisciplinary groups.	**N**	--
PO10	Ability to have verbal and written communication skills to use effectively not only with engineers but also with community at large.	**N**	--
PO11	Ought to have strong fundamentals in information technology and be able to have lifelong learning required for professional and individual developments.	**H**	Discussions, Exercises

PO12	Able to design, implement and manage projects in information technology with optimum financial recourses with, environmental awareness and safety aspects	**N**	**--**

N-NONE **S-SUPPORTIVE** **H-HIGHLY RELATED**

VIII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes		Level	Proficiency assessed by
PSO1	The ability to understand, analyze and develop computer programs in the areas related to algorithms, application development, multimedia, web design, big data analytics, and networking for efficient design of computer	**H**	Lectures, Assignments
PSO2	The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for industry success.	**H**	Projects
PSO3	The ability to employ modern computer languages, environments, platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.	**H**	Guest Lectures

N-None **S-Supportive** **H-Highly Related**

XI. SYLLABUS:

UNIT- I

Basic concepts- Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction Performance analysis - time complexity and space complexity, Asymptotic Notation-Big O, Omega and Theta notations
Introduction to Linear and non linear data structure

Singly Linked Lists-Operations-Insertion, Deletion, Concatenating singly linked lists, circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists-Operations- Insertion, eletion.

Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

UNIT- II

Stack ADT, definition, operations, array and linked implementations in C, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition and operations ,array and linked

Implementations in C, Circular queues-Insertion and deletion operations, Deque (Double ended queue)ADT, array and linked implementations in C.

UNIT- III

Trees – Terminology, Representation of Trees, Binary tree ADT, Properties of Binary Trees, Binary Tree Representations-array and linked representations, Binary Tree traversals, threaded binary trees, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

Graphs – Introduction, Definition, Terminology, Graph ADT, Graph Representations-Adjacency matrix, Adjacency lists, Graph traversals- DFS and BFS

UNIT- IV

Searching- Linear Search, Binary Search, Static Hashing-Introduction, hash tables, hash functions, Overflow Handling.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Comparison of Sorting methods.

UNIT- V

Search Trees-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion, AVL Trees-

Definition and Examples, Insertion into an AVL Tree ,B-Trees, Definition, B-Tree of order m, operations-Insertion and Searching, Introduction to Red-Black and Splay Trees(Elementary treatment-only Definitions and Examples),

Comparison of Search Trees Pattern matching algorithm- The Knuth-Morris-Pratt algorithm, Tries (examples only).

TEXT BOOKS

1. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan Anderson-Freed, Universities Press.
2. Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

REFERENCE BOOKS:

1. Data structures: A Pseudocode Approach with C, 2nd edition, R.F.Gilberg And B.A.Forouzan, Cengage Learning.
2. Data structures and Algorithm Analysis in C, 2nd edition, M.A.Weiss, Pearson.
3. Data Structures using C, A.M.Tanenbaum,Y. Langsam, M.J.Augenstein, Pearson.
4. Data structures and Program Design in C, 2nd edition, R.Kruse, C.L.Tondo and B.Leung,Pearson.
5. Data Structures and Algorithms made easy in JAVA, 2nd Edition, Narsimha Karumanchi, CareerMonk Publications.
6. Data Structures using C, R.Thareja, Oxford University Press.
7. Data Structures, S.Lipscutz, Schaum's Outlines, TMH.
8. Data structures using C, A.K.Sharma, 2nd edition, Pearson..
9. Data Structures using C &C++, R.Shukla, Wiley India.
10. Classic Data Structures, D.Samanta, 2nd edition, PHI.

X. COURSE PLAN:

The course plan is meant as a guideline. There may probably be changes.

Lecture No	Course Learning outcomes	Topics to be covered	Reference
1	Understand algorithms and its specification	Algorithm, pseudo code for expressing algorithm	T1:1.3
2-4	Understand space and time complexity and calculate performance	Space complexity, time complexity. Asymptotic Notations- Big Oh notation, Omega notation, Theta notation and little oh notation	T1:1.5
5-6	Understand recursive algorithms and data abstraction	Examples of recursions	T1:1.3.2
7-8	List types of data structures	Types of data structures	T1:1.4
9-12	Illustrate single linked list	Operations	T1:4.1
13-14	Illustrate circular linked lists	Operations like insertion, deletion	T2
15-17	Illustrate Double linked lists	Operations like insertion, deletion	T1:4.8
18-19	Apply concepts of arrays	One dimensional and two dimensional arrays	T1:2.1
20-21	Understand sparse matrix	Array and linked representation	T1:2.5
22-24	Identify stack and its operations	Stack definition, array and linked representation, stack applications	T1:3.1, 3.6.2
25-27	Identify queue and its operations	Queue ADT, array and linked representation, operations- insertion and deletion	T1:3.3
28-30	Illustrate circular queues	Insertion, deletion	T1:3.4
31-32	Understand Dequeue	Array and linked representation	T2
33-35	Elaborate trees and binary trees	Representation, Properties	T1:5.1, 5.2, 5.3
36-39	Understand priority queue and max heap	Priority queue ADT, insertion and deletion in max heap	T1:9.1
40-43	Elaborate graphs	Representation, graph traversal algorithms	T1:6.1
44-45	Describe searching techniques	Linear search and binary search	T2
46-50	Compare and contrast sorting techniques	Insertion sort, selection sort, radix sort, quick sort, heap sort and their comparison	T1:7.2

51-54	Understand hashing techniques	Hash tables, hash functions, overflow handling	T1:8.2, 8.3
55-60	Elaborate search trees	Binary search trees, AVL trees, Red, Black trees, Splay trees	T1:10.1, 10.2, 10.3, 10.4
61-63	Apply pattern matching algorithm	KMP algorithm	T2
64-65	Understand tries	Tries definition and example	T1:12.4

XI. MAPPING COURSE OBJECTIVES LEADING TO THE ACHIEVEMENTS OF THE PROGRAM OUTCOMES

Course objectives	Program outcomes												Program specific outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO2	PSO 3
1		H	H					S					H		
2			H								H				H
3			H	S				S			H		H		
4	H										H			H	
5		H													H

S-Supportive **H-Highly Related**

XII. MAPPING COURSE OUTCOMES LEADING TO THE ACHIVEMENT OF PROGRAM OUTCOMES PROGRAM SPECIFIC OUTCOMES:

Course outcomes	Program outcomes												Program specific outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
1		H	H					S					H		H
2	H		H								H			H	
3			H					S			H				
4		H											H	H	
5	H			S											H
6		H									H		H		H
7											H			H	
8	H			S											H

S-Supportive **H-Highly Related**

Prepared By: E.Soumya Asst.Professor **HOD-IT**

Date: 05-07-2015