COVER PAGE

THEORY OF COMPUTATION (Course Code)

II B.Tech: II Sem **L:4 T: P: C:**

Name of the Instructor(s): Nagendar Yamsani, G. Sunil Reddy

No. of Hours/week: 4

Total number of hours planned: 64

Pre-requisite

- C Programming
- Basic Programming Skill and Mathematical Logic

Learning Resources

- Textbooks, Lecture Notes

Name of the Text Book:

1. Hopcroft H.E. and Ullman J.D, "Introduction to Automata Theory Languages and Computation", Pearson Education
2. Sipser, "Introduction to Theory of Computation", 2nd Edition Thomson

REFERENCE BOOKS:

1. Daniel I.A. Cohen, "Introduction to Computer Theory", John Wiley.
2. John C Martin, "Introduction to languages and the Theory of Computation", TMH.
3. Theory of Computer Science – Automata languages and computation -Mishra and Chandrashekar

Reading materials :

1. Online Video links

Additional Resources (links etc)

1. https://nptel.ac.in/courses/106104028/
2. https://www.comp.nus.edu.sg/~sanjay/cs4232.html
3. http://www.cs.virginia.edu/~robins/cs3102/

How to Contact Instructor:

- **In-person office hours**: (Common for all instructors)
  - Students can meet, whenever we have free schedule during the college hours. Specifically on working Wednesday and Saturday during 3 p.m. to 4 p.m.

- **Online office hours: Time and how to access**
  - **Instructor:** Nagendar Yamsani
    - Email-ID : nagendar.y@srecwarangal.ac.in
    - Phone numbers: 9866572973
    - LMS : 9:00 pm to 11:00pm

  - **Instructor:** G. Sunil Reddy
    - Email-ID : sunil_g@srecwarangal.ac.in
    - Phone numbers: 9676561828
    - LMS : 7:00 pm to 9:00pm

Technology Requirements: (optional)

- Laptops for coding
- Programming Software
- Learning management System (Google classroom, etc.)

Overview of Course:

- **What is the course about: its purpose?**
  - In theoretical computer science and mathematics, the theory of computation is the branch that deals with how efficiently problems can be solved on a model of computation, using an algorithm
  - This is course intended for undergraduate students in computer science. This course will introduce various models of computation and study their power and limitations. Also explore the properties of the corresponding language classes defined by these models and the relations between them
- **What are the general topics or focus?**

  1. Finite-state machine
  2. Pushdown automata
  3. Linear-bounded automata
  4. Turing machine

- **How does it fit with other courses in the department or on campus?**

  This course introduces a formal framework for investigating both the computability and complexity of problems. It introduced various models of computation including finite automata, regular languages, context-free grammars, and Turing machines. These models provide a mathematical basis for the study of computability theory—the examination of what problems can be solved and what problems cannot be solved—and the study of complexity theory—the examination of how efficiently problems can be solved, including the P versus NP problem. This course helps us to fit its basic concepts to implement in Design of Analysis and Algorithms in evaluating the time complexity

- **Why would students want to take this course and learn this material?**

  1. Helps the student to improve problem solving skill.
  2. Helps in learning further programming languages.
  3. Helps to develop applications.
  4. As it a logical oriented, students will be able to improve logical thinking.

Methods of instruction

- Lecture (chalk & talk / ICT)
- Collaborative Learning (Think pair share)

Workload

- Estimated amount of time student needs to spend on course readings (per week): 2 hours per week

- Estimate amount of time to student needs to spend on Homework for practicing the problems (per week) : 2 Hours per week

Assessment

S. No	Assessments	Assessment Methodology	No of assessments	Weightage in marks	Marks scaled to
	CIE	Quizzes	2	5	2.5
		Class test	2	5	2.5
		Assignment	2	5	5
		Course Activity	--	--	--
		Course Project	--	--	--
		Internal exams	2	20	20
	SEE	--	--	--	70

Note:

- Class test/ Quiz – schedule:

Topic	Activity	Rubrics	UNIT	Schedule
Finite Automata	Quiz	10 Questions will be displayed one mark each	I	3rd week
Finite Automata	Assignment	10 Questions will be given one mark each	I	6th week
Summary of questions will be framed for mid1 syllabus	Class test	5 Questions will be given one mark each	Mid I Syllabus	8th week
Regular Grammar	Quiz	10 Questions will be displayed one mark each	IV	11th week
Context Free Grammar	Assignment	10 Questions will be given one mark each	IV	12th week
Summary of questions will be framed for mid2 syllabus	Class test	5 Questions will be given one mark each	Mid II Syllabus	16th week

- If the assessment is through online, the results will be displayed to the students immediately

Absentees for class assessments

In-time Assignments	5 marks
Late assignment within 5 days	4 Marks
Late Assignments even after 5 days	New set of questions will be given (with highest marks as 3). In case the assignment is not submitted in time then he/she will given zero marks

- Since the assessment is through online the results will be displayed to the students immediately.

// Absentees for Quiz

Absentees for Quiz	In case the student is absent then a set of problems will be given to execute using a code as an assignment with a deadline, in case the assignment is not submitted in time then he/she will given zero marks

Key concepts:

1. String
2. Language, Grammar
3. Finite-state machine
4. Pushdown automata
5. Linear-bounded automata
6. Turing machine

LESSON PLAN

Course Outcomes (COs):

At the end of the course the student should be able to:

1. Understand the basic concepts of Automata.
2. Apply grammars and Languages to Construct an Automata
3. Analyze the classification of Languages to Grammars, Languages and Automata.
4. Analyze normal forms, ambiguity grammars and simplification of grammars
5. Implement and submit a report on Automata Applications.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs)

Course Outcomes (COs) / Program Outcomes (POs)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**
Understand the basic concepts of Automata	3	2											1	
Apply grammars and Languages to Construct an Automata	3	3	3										3	3
Analyze the classification of Languages to Grammars, Languages and Automata.		2		2										2
Analyze normal forms, ambiguity grammars and simplification of grammars		2		2										
Implement and submit a report on Automata Applications.					2					3				2

Course Content (Syllabus)

UNIT- I

Fundamentals: Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automation and non deterministic finite automaton, transition diagrams.

UNIT- II

Finite Automata: NFA with Î transitions, acceptance of languages. Conversions and Equivalence: Equivalence between NFA with and without Î transitions, NFA to DFA conversion

Minimisation: Minimization of FSM, equivalence between two FSM's, Finite Automata with output-Moore and Mealy machines

UNIT-III

Regular Languages: Regular sets, regular expressions, identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets (proofs not required)

UNIT- IV

Grammar Formalism: Chomsky hierarchy of languages, Regular grammars-right linear and left linear grammars, equivalence between regular linear grammar and FA, inter conversion.

Context free grammar: Context free grammar, derivation trees, sentential forms. Right most and leftmost derivation of strings. Ambiguity in context free grammars. Minimisation of Context Free Grammars.

UNIT -V

CFG Normal Forms: Chomsky normal form, Greiback normal form, Enumeration of properties of CFL (proofs omitted)

Push Down Automata: Push down automata, definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state.

UNIT- VI

Turing Machine: Turing Machine, definition, model, design of TM, recursively enumerable languages, types of Turing machines (proofs not required).

Computability Theory: Linear bounded automata and context sensitive language, Posts Correspondence problem.

TEXT BOOKS

1. Hopcroft H.E. and Ullman J.D, "Introduction to Automata Theory Languages and Computation", Pearson Education.
2. Sipser, "Introduction to Theory of Computation", 2nd Edition Thomson

REFERENCES BOOKS

1. Daniel I.A. Cohen, "Introduction to Computer Theory", John Wiley.
2. John C Martin, "Introduction to languages and the Theory of Computation", TMH.
3. Theory of Computer Science – Automata languages and computation -Mishra and Chandrashekar

LESSON PLAN

S.No.	Topic	Delivery Method/ Activity
	UNIT I	
1	Introduction, Strings, Alphabet, Language, Operations	Chalk & Talk
2	Finite state machine, definitions	Chalk & Talk
3	finite automaton model	Chalk & Talk
4	acceptance of strings, and languages	Chalk & Talk
5	Deterministic Finite Automaton	Chalk & Talk
6	Non Deterministic Finite Automaton	Chalk & Talk
7	Transition Diagrams	Chalk & Talk
Quiz will be conducted on Finite Automata through Google classroom / Google forms		
	UNIT II	
8	NFA with Î transitions	Chalk & Talk
9	acceptance of languages for NFA	Chalk & Talk
10	Conversions and Equivalence	Chalk & Talk
11	Equivalence between NFA with and without Î transitions	Chalk & Talk
12	NFA to DFA conversion	Chalk & Talk
13	minimization of FSM	Chalk & Talk
14	minimization of FSM (alternate method)	Chalk & Talk
15	equivalence between two FSM’s	Chalk & Talk
16	Finite Automata with output- Moore Machines	Chalk & Talk
17	Finite Automata with output- Melay Machines	Chalk & Talk
18	Inter conversion of Moore and Melay Machines	Chalk & Talk
Assignment		
	UNIT III	
19	Regular Sets, Regular Expressions	Chalk & Talk
20	Identity Rules	Chalk & Talk
21	Constructing finite Automata for a given regular expressions	Chalk & Talk
22	Conversion of Finite Automata to Regular expressions	Chalk & Talk
23	Pumping lemma of regular sets	Chalk & Talk
24	closure properties of regular sets	Chalk & Talk
Class Test		
	I Mid Term Examinations	
	UNIT IV	

25	Chomsky hierarchy of languages	Chalk & Talk
26	Regular grammars	Chalk & Talk
27	right linear and left linear grammars	Chalk & Talk
28	equivalence between regular linear grammar and FA	Chalk & Talk
29	inter conversion of regular grammar and FA	Chalk & Talk
30	Context free grammar, derivation trees	Chalk & Talk
31	Sentential forms. Right most derivation of strings. Leftmost derivation of strings.	Chalk & Talk
32	Ambiguity in context free grammars	Chalk & Talk
33	Minimization of Context Free Grammars (introduction)	Chalk & Talk
34	Minimization of Context Free Grammars (elimination of useless symbols and productions)	Chalk & Talk
35	Minimization of Context Free Grammars (elimination of NULL productions)	Chalk & Talk
36	Minimization of Context Free Grammars (elimination of UNIT productions)	Chalk & Talk
Quiz will be conducted on Regular Grammar/ CFG through Google classroom / Google forms		
	UNIT V	
37	Chomsky normal form	Chalk & Talk
38	Greiback normal form	Chalk & Talk
39	Enumeration of properties of CFL	Chalk & Talk
Assignment		
40	Push down automata, definition, model	Chalk & Talk
41	acceptance of CFL	Chalk & Talk
42	Acceptance by final state	Chalk & Talk
43	acceptance by empty state and its equivalence	Chalk & Talk
	UNIT VI	
44	Turing Machine, definition, model, design of TM	
45	Computable functions	Chalk & Talk
46	recursively enumerable languages	Chalk & Talk
47	types of Turing machines	Chalk & Talk
48	linear bounded automata	Chalk & Talk
49	context sensitive language	Chalk & Talk
50	Post Correspondence Problem	Chalk & Talk
Class Test		
	II Mid Term Examinations	

OOPC Through Java (ES)

II B. Tech/II Sem **L: 3 T: - P: 6 C: 3**

Name of the Instructor(s): **T. Sampath Kumar, K. Ravi Chythanya, S. Tharun Reddy**

No. of Hours/week: 3

Total number of hours planned: 48

Pre-requisite

- Algorithms & Flowcharts
- Any Programming Language
- Mathematics Basics

Learning Resources:

1. Laptops with Class Work
2. Java Software
3. Mobiles with Internet Facility for successful completion of Online Quizzes.

Required Resources:

Text Books:

1. Java-The Complete Reference 9th Edition, Hebert Schlidt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

Reference Books:

1. An Introduction to Programming and OO Design using Java, J. Nino and F.A. Hosch, John Wiley & Sons.
2. Introduction to Java Programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, University Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd Edition, Oxfoord Univ. Press.
5. Java Programming and Object-Oriented application development, R.A. Johnson, Cengage Learning.

Reading Resources:

1. Lecture Notes
2. PPTs

Additional Resources:

Web Links:

1. https://www.tutorialspoint.com › java
2. https://www.javatpoint.com › java-tutorial
3. https://www.geeksforgeeks.org › java

How to Contact Instructor:

- **In-person office hours**:
  1. T. Sampath Kumar
     - Students can meet, whenever we have free schedule during the college hours.
     - Can meet 4:00 PM to 5:00 PM on Thursday and Friday.
  2. K. Ravi Chythanya
     - Students can meet, whenever we have free schedule during the college hours.
     - Can meet 4:00 PM to 5:00 PM on Monday and Tuesday.
  3. S. Tharun Reddy
     - Students can meet, whenever we have free schedule during the college hours.
     - Can meet 4:00 PM to 5:00 PM on Wednesday and Saturday.
- **Online office hours: time and how to access**
  1. T. Sampath Kumar
     - Email-ID : sampath_kumar_t@srecwarangal.ac.in
     - Phone numbers: 9581314677
  2. K. Ravi Chythanya
     - Email-ID : ravi_chythanya@srecwarangal.ac.in
     - Phone numbers: 9000188956
  3. S. Tharun Reddy
     - Email-ID : tarun_s@srecwarangal.ac.in
     - Phone numbers: 8106544849

Technology Requirements:

- JDK 1.8
- Google Classroom and Kahoot

Overview of Course:

- **What is the course about: its purpose?**
  Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C , but it has fewer low-level facilities than either of them. Java was one of the most popular programming languages in use according to GitHub, particularly for client-server web applications, with a reported 9 million developers.
- **What are the general topics or focus?**
  - ➢ OOP Concepts,
  - ➢ Handling errors –Exceptions,
  - ➢ Multi-Threaded Programs,
  - ➢ File Handling
  - ➢ GUI Development.
- **How does it fit with other courses in the department or on campus?**
  - ➢ Web Technologies,
  - ➢ Mobile Application Development,
  - ➢ Project Work
- **Why would students want to take this course and learn this material?**
  - ➢ Popularity and High Salary
  - ➢ Powerful Development Tools
  - ➢ Java has a Large Community
  - ➢ Java is Versatile
  - ➢ Multiple Open Source Libraries
  - ➢ Enhance their Programming Knowledge
  - ➢ Own Idea Implementation using GUI Applications(swings)

Methods of instruction

- Lecture (chalk & talk / ICT)
- Collaborative Learning (Role Play, Group Activity)

Workload

- Estimated amount of time student needs to spend on course readings (per week) -3
- Estimate amount of time to student needs to spend on course assignments and projects (per week) -6

Assessment

S. No	Assessments	Assessment Methodology	No of assessments	Weightage in marks	Marks scaled to
1	CIE	Quizzes	2	5	2.5
2		Class test	2	10	2.5
3		Assignment	--	--	--
4		Course Activity	4	10	5
5		Course Project	--	--	--
6		Internal exams	2	20	20
7	SEE	--	--	--	70

Class test/ Quiz:
Schedule:

Test-Type	Syllabus	Tentative Date & Time	Mode
Class Test-1	1st Unit	3rd Week-Last Working Hour	Offline
Quiz-1	1st Unit and 2nd Unit	8th Week-Last Working Hour	Online
Quiz-2	3rd Unit and 4th Unit	14th Week-Last Working Hour	Online
Class Test-2	5th Unit and 6th Unit	16th Week-Last Working Hour	Offline

Activities:

Topic	Activity	Rubrics			Unit	Schedule
Structured Programming Vs OOP Concepts	Group Discussion	Problem Statement(5)	Design (5)	Total (10)	I	4th Week
Inheritances and their Importance	Think Pair Share Activity	NIL			II	8th Week
Error Handling and Types	Jigsaw	NIL			III	12th Week
Threads	Role Play	NIL			IV	16th Week

- Grades (will be shared immediately if its online and within 3 days from the activity if it is offline)

Grade	Marks Range
Grade ‘A’	>=8
Grade ‘B’	>=5 and <8
Grade ‘C’	>=3 and <5
Grade ‘D’	<3

- Absentees for class assessments:
  - With HOD permission Re-conduction of the Class Assessment will be done within next two working days from 4:00 PM to 5:00PM, Class Test/Quiz mark is evaluated for 75% of original marks.
  - Students who have taken prior permission from the HOD, re-conduction of the Class Assessment will be done within next two working days from 4:00 PM to 5:00PM Class Test/Quiz mark is evaluated for original marks.

Key concepts:

- OOP features and their implementations,
- Exception Handling,
- Multi-threading

Difficult Topics:

- I/O Streams,
- Collections Framework,
- GUI Programming.

LESSON PLAN

Course Outcomes (COs):

At the end of the course the student should be able to:

1. Identify classes, objects, members of a class and relationships among them needed for a specific problem.
2. Demonstrate the concept of polyorphism, inheritance and re-usability.
3. Illustrate Java Programs to implement error handling techniques using exception handling.
4. Compare Multithreaded programming with ordinary programming models, file handling techniques.
5. Build GUI interface using Collections.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs)

Course Outcomes (COs) / Program Outcomes (POs)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**
CO1	2	3	2	2	3	1							2	2
CO2			3					1	2			2	1	2
CO3				3	2		2				2		2	3
CO4				3			1		1	2		2	1	3
CO5			3		2	3						2	2	3

Course Content (Syllabus)

UNIT I

Object-oriented thinking- A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

UNIT II

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance.

Packages- Defining a Package, CLASSPATH, Access protection, importing packages. Command Line Arguments.

UNIT III

Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

UNIT IV

Stream based I/O(java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

UNIT V

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hash table, Properties, Stack, Vector More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT VI

GUI Programming with Swing:

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

A Simple Swing Application, **Applets** – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs. Layout Mangers.

LESSON PLAN

Lecture No.	Topic	Delivery Method/ Activity
UNIT – I		
L1	**Object- oriented Thinking-**A way of viewing world- Agents and Communities, messages and methods, Responsibilities	Chalk & Talk/ PPT
L2	Classes and Instances	Chalk & Talk/ PPT
L3	Class Hierarchies- Inheritance, Method binding	Chalk & Talk/PPT
L4	Over riding and Exceptions, Summary of Object-Oriented Concepts	Chalk & Talk/PPT
L5	Java Buzz Words, An Overview of Java, Data Types, Variables and Arrays	Chalk & Talk/PPT
L6	Operators, expressions, control Statements	Chalk & Talk/PPT
L7	Introducing classes, Methods and classes	Chalk & Talk/PPT
L8	String handling, Structured Programming Vs OOP Concepts	Chalk & Talk, **Activity:** Group Discussion
UNIT – II		
L9	**Inheritance-** Inheritance Concept, Inheritance Basics, Member access, Constructors	Chalk & Talk/PPT
L10	Creating Multi level hierarchy, super use, using final with inheritance.	Chalk & Talk/PPT
L11	Polymorphism- adhoc polymorphism, pure polymorphism, method overriding.	Chalk & Talk/PPT
L12	Abstract Classes, Object Class, Inheritance and their Importance	PPT, **Activity:** Think-Pair-Share
L13	Forms of inheritance- specialization, specification, construction, extension	Chalk & Talk/PPT
L14	Limitation, combination, benefits of inheritance. **Packages-** Defining a package.	Chalk & Talk/PPT
L15	Class Path, Access Protection, importing packages, Command Line arguments.	Chalk & Talk/PPT
UNIT – III		
L16	**Interfaces-** Defining Interface, implementing interfaces	Chalk & Talk/PPT
L17	Nested Interfaces, applying interfaces, variables in interfaces and extending interfaces	Chalk & Talk/PPT
L18	**Exception Handling-** Fundamentals of exception handling, Exception types, Termination of presumptive models,	Chalk & Talk/PPT

	Uncaught Exceptions	
L19	Using try and catch, multiple catch clauses	Chalk & Talk/PPT
L20	Nested try statements, throw, throws and finally, Error Handling and Types	PPT, **Activity:** Jigsaw
L21	Built-in exceptions, creating own exception sub classes	Chalk & Talk/PPT
UNIT – IV		
L22	**Stream Based I/O (java. io)-** The Stream classes- Byte Streams and character streams	Chalk & Talk/PPT
L23	Reading console Input and Writing Console Output, File class	Chalk & Talk/PPT
L24, L25	Reading and Writing Files, Random access file operations	Chalk & Talk/PPT
L26	The Console class, Serialization	Chalk & Talk/PPT
L27	Enumerations, auto boxing, generics	Chalk & Talk/PPT
L28	**Mutli- Threading-** Differences between thread-based multitasking and process based multi tasking , Java Thread Model	Chalk & Talk/PPT
L29	Creating Threads, Thread Priorities	Chalk & Talk/PPT
L30	Synchronizing Threads	Chalk & Talk/PPT
L31	Inter Thread Communication, Threads & Streams	Chalk & Talk/PPT, **Activity:** Flipped Class Room
UNIT – V		
L32	**The Collection Frame work (java.util)-** Collections overview, Collections Interfaces	Chalk & Talk/PPT
L33, L34	The Collections Classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array DeQue, Accessing collection via Iterator, For-each Alternative,	Chalk & Talk/PPT
L35	Map Interfaces and Classes, Comparators	Chalk & Talk/PPT
L36	Collection Algorithms- Arrays, The Legacy classes and Interface- Dictionary, Properties	Chalk & Talk/PPT
L37	Stack, Vector More Utility Classes, String Tokenizer	Chalk & Talk/PPT
L38	Bit Set, Date, Calendar	Chalk & Talk/PPT
L39	Random, Formatter, Scanner	Chalk & Talk/PPT
UNIT- VI		
L40	**Event Handling-** Delegation event model, Events, Event sources, Event classes, Event Listeners.	PPT

L41	Handling mouse and keyboard events	PPT
L42	Adapter classes, inner classes, Anonymous Inner Classes	PPT
L43, L44	**A Simple Swing Application, Applets-** Applets and HTML, Security Issues, Applets and Applications, Passing Parameters to Applet	PPT
L45	Exploring Swing Controls- JLabel and Image Icon, JText Field	PPT
L46	The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JSrcoll Pane	PPT
L47	Jlist, JCombo Box	PPT
L48	Swing Menus, Dialogs, Layout Managers	PPT

COVER PAGE

DATABASE MANAGEMENT SYSTEMS (Course Code)

II B.Tech: II Sem **L:3 T:1 P:- C:4**

Name of the Instructor(s): Ch. Sandeep, Syed Nawaaz Pasha

No. of Hours/week: 4

Total number of hours planned: 54

Pre-requisite

- Knowledge/skills needed to succeed in this course: Logical thinking and Analyzing abilities

Learning Resources:
Course notes, PPTs, Web links

Required Resources:

Name of the Textbook(s):

1. Raghurama Krishnan, Johannes Gehrke“ Data base Management Systems” TATA McGrawHill 3rd Edition.
2. Silberschatz, Korth “Data base System Concepts” McGraw hill, V Edition.

Reading materials:

1. Peter Rob and Carlos Coronel “Data base Systems design, Implementation, and Management” 7th Edition.
2. Elmasri Navrate “Fundamentals of Database Systems” Pearson Education.

Additional Resources (links etc):

1. http://nptel.iitm.ac.in
2. http://highered.mheducation.com/sites/0072465638/student_view0/index.html

How to Contact Instructor:

Sandeep Chintham:

- In-person office hours: 9:30 AM to 5:00 PM – Room no.: 1311, except class timings
- Online office hours: 9:30 AM to 5:00 PM - Except class timings, a mail or message
  - Mail: sandeep_ch@srecwarangal.ac.in
  - Phone numbers: 9985307469
- Other than office hours: A message to the above number from 6PM to 9PM from Monday to Saturday and 8AM to 10 AM on Sunday

Syed Nawaz Pasha:

- In-person office hours: 9:30 AM to 5:00 PM – Room no.: 1311, except class hours
- Online office hours: 9:30 AM to 5:00 PM - Except class timings, a mail or message
  - Mail: syed_nawaz_pasha2srecwarangal.ac.in
  - Phone numbers: 9959054349
- Other than office hours: A message to the above number from 6PM to 9PM from Monday to Saturday.

Technology Requirements:

- Laptops for class work only on required days
- MySQL or Oracle software
- Google classroom

Overview of Course:

- What is the course about: its purpose?
  Being a Computer Science and Engineering student, one should learn a course on Database Management Systems because, creating and managing a database is ever needed phenomenon for all the organizations.
  A database management system (DBMS) is system software for creating and managing databases. A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.
  The DBMS is perhaps most useful for providing a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner. A DBMS can limit what data the end user sees, as well as how that end user can view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage media it resides because the DBMS handles all requests.

- What are the general topics for focus?
  1. The student has to focus on understanding the structure of a DBMS, so that they can develop an effective database in the lab or during course project.
  2. To develop an effective database, E-R diagrams play a vital role, students should concentrate on developing better E-R Diagrams
  3. Since a developed database should be free from inconsistencies or anomalies, the student is required to focus more on normalization techniques and normal forms
  4. Extracting data from an existing database requires knowledge of SQL queries, hence the student should concentrate on all SQL commands

- How does it fit with other courses in the department or on campus?
  As a student learns few programming languages like C-programming, Object Oriented Programming through Java etc. in the first and second years of engineering, he/she will be able to develop a front end for any web based applications. Now by learning this course, the student will be able to develop a backend database for any web based application. By integrating the front end and backend applications, the student can develop a complete web application for a given real time problem

- Why would students want to take this course and learn this material?
  1. It is very important for a Computer Science and Engineering student to know in detail about Database management system, its creation, modification and its functionality.
  2. Good number of questions are asked in all competitive exams from this subject
  3. Placement interviews and Placement exams also include questions from this subject

Methods of instruction

- Lecture using PPTs in classroom
- Brainstorming and Discussion
- Group work as pair or sometimes as a group of 4 students
- Activities like Role plays, Think-Pair-Share
- Flipped classroom teaching
- Chalk and Talk while solving problems in some topics

Workload

- Estimated amount of time to spend on course readings
  Students are informed to spend half an hour per day (any four days of a week) or maximum of two hours per week on course readings
- Estimate amount of time to spend on course assignments and projects
  One assignments is given during the delivery of this course. Students will need to spend couple of hours per day for a maximum of three days or one hour per day for a maximum of six days to finish the assignment
- One course project should be done by a group of students (maximum 4 students in a group). Students need to spend one hour per day for a maximum of one week to finish the course project

Assessment:

S. No	Assessments	Assessment Methodology	No of assessments	Weightage in marks	Marks scaled to
1	CIE	Quizzes	2	10	2.5
2		Class test	--	--	--
3		Assignment	1	10	5
4		Course Activity	2	20	2.5
5		Course Project	1	10	5 (In Lab)
6		Internal exams	2	20	20
7	SEE	--	--	--	70

Note:

- Class test/Quiz – Quiz in week 6 and 14
- Grades (will be shared immediately if its online and within 3 days from the activity if it is offline)

Topic	Activity	Rubrics			UNIT	Schedule
E-R Diagram	Group Activity	Problem Statement(10)	Design (10)	Total (20)	I	4th Week
Normal forms	Think Pair Share Activity	Summary of the topic with Multiple Choice Questions (With wrong answers) and later with the discussion related to wrong answers			III	10th Week
Relational Algebra	Online Assessment	10 Questions will be displayed one mark each			II	6th Week
Transaction management and Normalization	Online Assessment	10 Questions will be displayed one mark each				11th Week
Relational Model	Group Activity	Problem Statement(10)	Design (10)	Total (20)	I	5th Week

- If any student is absent for class assessments with prior permission, the retest will be conducted on a later intimated date
- Any late submission of assignment, the student will be awarded zero marks
- Any student absent for online test will be awarded zero marks

Key concepts:

1. File system vs Database system
2. Database structure
3. Designing E-R models
4. Designing Relational models
5. Enforcing Integrity Constraints
6. Normalization and Normal forms
7. Transaction management
8. Recoverability management
9. Storage and Indexing

LESSON PLAN

Course Outcomes (COs):

At the end of the course, the students will develop ability to

1. Understand basic concepts of DBMS.
2. Apply Data Models to perform various operations on database.
3. Analyze normalization techniques for the given database application.
4. Evaluate the concepts of transaction, Concurrency and Recovery techniques in database.
5. Design a Data model for real time problem.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs)

Course Outcomes (COs) / Program Outcomes (POs)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**
CO1	2	3	1										2	
CO2		2	3	3										2
CO3		2	3	3										
CO4			2	3									2	
CO5			3	3									3	

Course Content (Syllabus)

Lecture No.	**Topic**	**Delivery Method/ Activity**
Unit-I		
1	Introduction to DBMS	Brainstorming
2	Advantages and Disadvantages of File systems with examples	Brainstorming, PPT
3	Data base System VS file System	PPT
4	View of Data, Data Abstraction	Chalk and Talk
5	Instances and Schemas, data Models	Chalk and Talk
6	data Models	PPT, Chalk and Talk
7	Database Languages	PPT, Chalk and Talk
8	Data base System Structure	PPT
9	Data base design and ER diagrams	PPT, Chalk and Talk
10	Data base design and ER diagrams	PPT, Chalk and Talk

11	Beyond ER Design Entities	Chalk and Talk
12	Attributes and Entity sets	PPT, Chalk and Talk
13	Relationships and Relationship sets	PPT, Chalk and Talk
14	Additional features of ER Model	PPT, Chalk and Talk
15	Concept Design with the ER Model	PPT, Chalk and Talk
16	Design of ER Diagrams	Group Activity
Unit-II		
17	Introduction to the Relational Model	PPT, Chalk and Talk
18	Enforcing Integrity constraints	PPT, Chalk and Talk
19	Querying Relational Data	PPT, Chalk and Talk
20	Querying Relational Data, Logical data base Design	PPT, Chalk and Talk
21	Design Relational database	Group Activity
22	Introduction to Views, Destroying /altering Tables and Views	Chalk and Talk
23	Relational Algebra – Selection and projection operations	PPT, Chalk and Talk
24	set operations	PPT, Chalk and Talk
25	renaming, Joins, Division	PPT, Chalk and Talk
26	Examples of Algebra overviews, Quiz	Online test
27	Relational calculus, Tuple relational Calculus	PPT, Chalk and Talk
28	Domain relational calculus	PPT, Chalk and Talk
Unit-III		
29	Introduction to Schema Refinement - Problems Caused by redundancy	PPT, Think-Pair-Share
30	Decompositions- Problem related to decomposition, Functional Dependencies- Reasoning about FDS	PPT, Think-Pair-Share
31	Normal Forms - FIRST, SECOND Normal forms	PPT, Chalk and Talk
32	THIRD Normal form, BCNF	PPT, Chalk and Talk
33	Properties of Decompositions - Loss less join	PPT, Chalk and Talk
34	Decomposition, Dependency preserving Decomposition, FOURTH Normal form	PPT, Chalk and Talk
Unit-IV		
35	Transaction Concept- Transaction State	Chalk and Talk

36	Implementation of Atomicity and Durability	PPT, Chalk and Talk
37	Transaction management and Normal forms, Quiz	Online Test
38	Concurrent Executions – Serializability	PPT, Chalk and Talk
39	Concurrent Executions – Serializability	PPT, Chalk and Talk
40	Recoverability – Testing for serializability	PPT, Chalk and Talk
41	Recoverability – Testing for serializability	PPT, Chalk and Talk
Unit-V		
42	Lock Based Protocols	PPT, Chalk and Talk
43	Lock Based Protocols	PPT, Chalk and Talk
44	Timestamp Based Protocols	PPT, Chalk and Talk
45	Validation Based Protocols	PPT, Chalk and Talk
46	Recovery and Atomicity	PPT, Chalk and Talk
47	Log Based Recovery	PPT
48	Log Based Recovery	PPT
49	Recovery with Concurrent Transactions, Buffer Management	PPT
Unit-VI		
50	Overview of Storage and Indexing	Brainstorming, PPT
51	Data on External Storage	PPT
52	File Organization and Indexing	PPT
53	Hash Based Indexing, Tree base Indexing	PPT
54	Revision	--

COVER PAGE

Data Structures (Course Code)

II B.Tech: II Sem	**L:4**	**T:**	**P:6**	**C:4**

Name of the Instructor(s): Dr.M.Sheshikala,Dr.R.Vijaya Prakash, K.Sudheer Kumar

No. of Hours/week: 4

Total number of hours planned: 4

Pre-requisite

- C Programming
- Programming Skills

Learning Resources

- Laptops for class work
- C-Software
- Sometimes Mobiles to perform Activities .

Name of the Textbook :

1. Ellis Horowitz, Sartaj Sahani, Dinesh Metha, “Fundamentals of data Structures in C++”, Galgotia Publications Pvt. Ltd., ISBN 81-7515-27, 2003.
2. Mark Allen Weiss, “Data structure and algorithm analysis in C++”, 2nd Edition, Pearson Education, ISBN 81-2808-670-0.

REFERENCE BOOKS:

1. Herbert Schildt, “C++, The Complete Reference”, TMH, 4th Edition, ISBN: 9780070532465.
2. D. Samanta, “Classic Data Structures”, Prentice Hall India, ISBN 81-203-1874-9, 2002.

Reading materials :

1. Lecture notes.
2. Online Video links.

Additional Resources (links etc)

1. THE ART OF COMPUTER PROGRAMMING (Volume 1 / Fundamental Algorithms), Donald Knuth
2. Introduction to Algorithms, Cormen, Leiserson, Rivestand Stein

3. http://www.geeksforgeeks.org/data-structures/
4. https://www.slideshare.net/dharshanakrishnamurthy1/linked-list-ppt

How to Contact Instructor:

- **In-person office hours**:

  1. Dr.M.Sheshikala
     - Students can meet, whenever we have free schedule during the college hours.
     - Can meet 4:00 pm to 5:00 pm on Thursday and Friday.
  2. Dr. R. Vijaya Prakash

     - Students can meet, whenever we have free schedule during the college hours.
  3. K.Sudheer Kumar:
     - Students can meet, whenever we have free schedule during the college hours.
     - Can meet 4:00 pm to 5:00 pm on Thursday and Friday.

- **Online office hours: time and how to access**

  Dr.M.Sheshikala
  - Email-ID : sheshikala_m@srecwarangal.ac.in
  - Phone numbers: 9849116219
  - LMS : 9:00 pm to 10:00pm

  Dr.R. Vijaya Prakash:
  - Email-ID : vijaya_prakash_r@srecwarangal.ac.in
  - Phone numbers: 9951332996
  - LMS : 9:00 pm to 10:00pm

  K.Sudheer Kumar:
  - Email-ID : vijaya_prakash_r@srecwarangal.ac.in
  - Phone numbers: 9951332996
  - LMS : 9:00 pm to 10:00pm

Technology Requirements: (optional)

- Laptops for class work
- C-Software
- Learning management system (Google classroom / Kahoot)

Overview of Course:

- **What is the course about: its purpose?**

  - In our day today life, everything is about data.
  - Yes, we have lots of data to play with, but to do that we need a proper place to store it and use it back.

- For example, we cannot store water in a bag, we store it in a bottle and we can't put vegetables or eggs in a bottle, we use a bag.
- Every container is designed to store specific items (here different data types/formats).
- So teaching data structure helps us to store the data we have in an efficient manner to retrieve it with low cost and less time.
- After using the efficient data structure, we can extract the information we want or process it for further analysis.

The course will introduce the problem solving using programs and design of algorithms and their complexity. It will review elementary data structures such as Arrays, Stack, Queues, Linked List, and related algorithms for manipulating the data structures. It will also discuss sorting and searching techniques, and their complexity. We will also briefly explore more advanced data structures such as graphs and graph algorithms, balanced trees, and heaps.

- **What are the general topics or focus?**

  1. Sorting's
  2. Searching's
  3. Stack & Queue using Arrays and Linked List
  4. Trees and Graphs
  5. Hashing

- **How does it fit with other courses in the department or on campus?**

  A good algorithm usually comes together with a set of good data structures that allow the algorithm to manipulate the data efficiently. In this course, we consider the common data structures that are used in various computational problems. We will learn how these data structures are implemented in different programming languages and will practice implementing them in our programming assignments. This will help us to understand what is going on inside a particular built-in implementation of a data structure and what to expect from it. This course helps us to fit its basic concepts to implement in Design of Analysis and Algorithms in evaluating the time complexity.

- **Why would students want to take this course and learn this material?**

  1. Helps the student to improve problem solving skill.
  2. Helps in learning further programming languages.
  3. Helps to develop applications.
  4. As it a concept oriented language, students will be able to improve logical thinking.
  5. Helps in understanding System Software's like Operating System.

Methods of instruction

- Lecture (chalk & talk / ICT)
- Collaborative Learning (Role Play, Group Activity)

- Few Activities

Workload

- Estimated amount of time student needs to spend on course readings (per week): 2 hours per week
- Estimate amount of time to student needs to spend on course assignments and projects (per week) : 3-4 Hours per week

Assessment

S. No	Assessments	Assessment Methodology	No of assessments	Weightage in marks	Marks scaled to
	CIE	Quizzes	2	5	5
		Class test	--	--	--
		Assignment	--	--	--
		Course Activity	2	10	5
		Course Project	--	--	--
		Internal exams	2	20	20
	SEE	--	--	--	70

Note:

- Class test/ Quiz – schedule to be specified

Topic	Activity	Rubrics	UNIT	Schedule
Sorting’s	Role Play	NIL	I	3rd Week
Linked List	Online Quiz	10 Questions will be displayed one mark each (10)	III	9th Week
Trees	Online Quiz	10 Questions (10)	IV	12th Week
Average		Scaled to 5 Marks		

- Since assessment is through online, the results will be displayed to the students immediately.

// Absentees for class assessments (Define Ground Rules)

Absentees for Quiz	In case the student is absent then a structured enquiry problem will be given as an assignment with a deadline, in case the assignment is not submitted in time then he/she will given zero marks.

Key concepts :

1. Sorting’s
2. Searching’s
3. Stack & Queue using Arrays and Linked List
4. Trees and Graphs
5. Hashing

LESSON PLAN

Course Outcomes (COs):

At the end of the course the student should be able to:

1. Ability to define, understand and explain basic concepts of Data Structures.
2. Ability to apply the concepts of Data Structures using Static and Dynamic Memory Allocation for solving real time problems
3. Ability to Analyse the performance of various Data Structures
4. Ability to choose an effective documentation on data structures.
5. Ability to Develop and Submit a report on real world problems.

Course Articulation Matrix: Mapping of Course Outcomes (COs) with Program Outcomes (POs)

Course Outcomes (COs) / Program Outcomes (POs)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**PSO1**	**PSO2**
. Explain how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms.	3	3	2	-	1	3	-	1	2	-	2	3	3	-
Compare and contrast the benefits of dynamic and static data structures implementations	3	3	2	-	1	3	-	1	2	-	2	3	3	-
Develop and Evaluate programs that use arrays, records, linked structures, stacks, queues, trees and graphs	3	3	2	-	1	3	-	1	2	-	2	3	3	-
Demonstrate different methods for traversing trees	3	3	2	-	1	3	-	1	2	-	2	3	3	2
Design and implement an appropriate hashing function for an application	3	3	2	-	1	3	-	1	2	-	2	3	3	-

Course Content (Syllabus)
UNIT- I

Basic concepts - Data types, Abstract Data Types, Data structures, Algorithms.

Searching- Linear Search, Binary Search

Sorting- Bubble Sort, Insertion Sort, Selection Sort, Quick sort, Merge sort, Comparison of Sorting methods.

UNIT- II

Stack ADT - Definitions, operations, array and linked implementations, applications-infix to postfix conversion, recursion implementation,

Queue ADT - Definitions and operations, array and linked Implementations, Applications of Queue Circular queues and operations

UNIT-III

Linear data structures - Linear Lists, Sequential and Linked allocation ,The list ADT, array and linked Implementations, Singly Linked Lists-Operations-Insertion, Deletion, Doubly Linked Lists-Operations - Insertion, Deletion

UNIT- IV

Non Linear data structures: Trees – Basic Terminology, Binary tree ADT, array and linked representations, traversals, threaded binary trees, Disjoint Sets, Union and Find algorithms, Priority Queues-Definition, ADT, Realizing a Priority Queue using Heap.

Search Trees-Binary Search Trees, Definition, ADT, Implementation, Operations-Searching, Insertion and Deletion, **AVL Trees** - Definition, Operations – Insertion and Searching,

B-Trees - Definition, B-Tree of order m, operations - insertion and deletion, Introduction to Red-Black and Splay Trees, Comparison of Search Trees.

UNIT -V

Graphs – Introduction, Basic Terminology, Graph Representations- Adjacency matrix, Adjacency lists, Adjacency multilists, Graph traversals- DFS and BFS, Spanning Trees – Kruskals, prims algorithms.

UNIT- VI

Hashing - hash table representation, hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

TEXT BOOKS

1. Mark Allen Weiss, “Data structures and Algorithm Analysis”, 3rd edition, Pearson Education. Ltd.,
2. S.Sahani, “Data structures, Algorithms and Applications”, Universities Press.

REFERENCE BOOKS

1. Michael T.Goodrich, R.Tamassia and D.Mount, “Data structures and Algorithms”, Wiley student edition, seventh edition, John Wiley and Sons.
2. Adam Drozdek, “Data structures and algorithms”, 3rd Edition, Cengage Learning.
3. Langsam, Augenstein and Tanenbaum, “Data structures using C”, PHI.
4. G.L.Heileman, “Data structures, algorithms and OOP”, TMH edition

LESSON PLAN

S.No.	Topic	Delivery Method/ Activity
UNIT – I		
1	Introduction, Bubble Sort	Chalk & PPT
2	Sorting: Insertion sort	Chalk & PPT
3	Quick sort	Chalk & PPT
4	Merge sort	Role Play Activity & PPT
5	Selection Sort	Chalk & PPT
6	Linear Searching	Chalk & PPT
7	Binary Searching	Chalk & PPT
8	**Activity : Comparison of Sorting Techniques using a Role Play**	Activity through Role Play
UNIT – II		
9	**Data structures:** Definition, Types, Abstract Data Type(ADT)	Chalk & PPT
10	**Stack**: Model, Representation using arrays	Chalk & PPT
11	Queue Types: Linear	Chalk & PPT
12	Circular Queue	Chalk & PPT
13	DEQueue - Model, Representation using arrays	Chalk & PPT
14	Operations, Applications	Chalk & PPT
UNIT – III		
15.	Linked List Introduction	Chalk and PPT

16.	Singly Linked Lists -Operations-Insertion, Deletion,	Chalk and PPT
17.	Double Linked Lists -Operations-Insertion, Deletion	Chalk and PPT
18.	Circular Singly Linked Lists -Operations-Insertion, Deletion	Chalk and PPT
19.	Circular Double Linked Lists -Operations-Insertion, Deletion	Group Activity through distribution of Problems to different groups
20.	Traversals	Chalk and PPT
21.	Stack implementation using pointers	Chalk and PPT
22.	Queue implementation using pointers	Chalk and PPT
23.	**Actitvity**	**Online Quiz Using Kahoot APP (On I & II Units)**
I Mid Examination		
UNIT – IV		
24.	**Non Linear Data Structures-** Trees – Basic Terminology	Chalk and PPT
25.	Binary tree	Chalk and PPT
26.	Binary search tree	Chalk and PPT
27.	Representation, creation	Chalk and PPT
28.	insertion and deletion operations, traversals	Chalk and PPT
29.	Balanced Trees: AVL	Chalk and PPT
30.	Balanced Trees: AVL	Chalk and PPT
31.	B-Trees – representation, Creation, insertion and deletion operations, traversals	Chalk and PPT
32.	RedBlack Tree, Creation, insertion and deletion operations, traversals	Chalk and PPT
33.	Splay Tree, Creation, insertion and deletion operations, traversals	Chalk and PPT
34.	**Activity**	**Online Quiz Using Kahoot APP**
UNIT V		
35	**Non Linear Data Structures,** Graphs: Basic Terminology	Chalk and PPT
36	graph representation &Implementation	Chalk and PPT
37	Graph Traversals: Depth first search Introduction and Program Logic	Chalk and PPT
38	Graph Traversals: Breadth first search Introduction and Program Logic	Chalk and PPT
39	Graph Traversals: Comparison of Depth first search & Breadth first search	Chalk and PPT

40	Minimum cost spanning tree Introduction	Chalk and PPT
41	Prim’s algorithms.	Chalk and PPT
42	Minimum cost spanning trees, Kruskal’s algorithms.	Chalk and PPT
UNIT-VI		
44	Hashing: Hash functions & methods	Chalk and PPT
45	Implementation of folding method	Chalk and PPT
46	Probing: quadratic probing,	Chalk and PPT
47	Double hashing	Chalk and PPT
48	Collision resolution	Chalk and PPT
49	Application of Data Structures in Computer Science and Engineering	Activity: Discussion
	II – Mid Examinations	