CHARUSAT

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Faculty of Technology & Engineering

Bachelor of Technology Programme

Information Technology

(B.Tech. IT)

ACADEMIC REGULATIONS & SYLLABUS

(Choice Based Credit System)

Faculty of Technology and Engineering

B. Tech. Programme

(Information Technology)

Chandubhai S. Patel Institute of Technology

Devang Patel Institute of Advance Technology and Research

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF TECHNOLOGY AND ENGINEERING

ACADEMIC REGULATIONS
Bachelor of Technology Programmes
Choice Based Credit System

To ensure uniform system of education, duration of undergraduate and post graduate programmes, eligibility criteria for and mode of admission, credit load requirement and its distribution between course and system of examination and other related aspects, following academic rules and regulations are recommended.

1. System of Education

Choice based Credit System with Semester pattern of education shall be followed across The Charotar University of Science and Technology (CHARUSAT) both at Undergraduate and Master's levels. Each semester will be at least 90 working day duration. Every enrolled student will be required to take a course works in the chosen subject of specialization and also complete a project/dissertation if any. Apart from the Programme Core courses, provision for choosing University level electives and Programme/Institutional level electives are available under the Choice based credit system.

2. Duration of Programme

(i)	Undergraduate programme	(B. Tech.)
	Minimum	8 semesters (4 academic years)
	Maximum	16 semesters (8 academic years)

3. Eligibility for admissions

As enacted by Govt. of Gujarat from time to time.

4. Mode of admissions

As enacted by Govt. of Gujarat from time to time.

5. Programme structure and Credits

As per annexure – I attached

6. Attendance

6.1 All activities prescribed under these regulations and listed by the course faculty members in their respective course outlines are compulsory for all students pursuing the courses. No exemption will be given to any student from attendance except on account of serious personal illness or accident or family calamity that may genuinely prevent a student from attending a particular session or a few sessions. However, such unexpected absence from classes and other activities will be required to be condoned by the Dean/Principal.

6.2 Student attendance in a course should be Minimum 80%.

7 Course Evaluation

7.1 The performance of every student in each course will be evaluated as follows:

7.1.1 Internal evaluation by the course faculty member(s) based on continuous assessment, for 30% of the marks for the course; and

7.1.2 Final examination by the University through written paper or practical test or oral test or presentation by the student or a combination of any two or more of these, for 70% of the marks for the course.

7.2 *University Examination*

7.2.1 The final examination by the University for 70% of the evaluation for the course will be through written paper and 100% for practical test or oral test or presentation by the student or a combination of any two or more of these.

7.2.2 **In order to earn the credit in a course a student has to obtain grade other than FF.**

7.3 Performance at Internal & University Examination

7.3.1 Minimum performance with respect to internal marks as well as university examination will be an important consideration for passing a course. Details of minimum percentage of marks to be obtained in the examinations (internal/external) are as follows

Minimum marks in University Exam per subject	Minimum marks Overall per subject
40%	45%

7.3.2 A student failing to score 45% of the final examination will get a FF grade.

7.3.3 If a candidate obtains minimum required marks per subject but fails to obtain minimum required overall marks, he/she has to repeat the university examination till the minimum required overall marks are obtained.

8 Grading

8.1 The total of the internal evaluation marks and final University examination marks in each course will be converted to a letter grade on a ten-point scale as per the following scheme:

Table: Grading Scheme (UG)

Range of Marks (%)	≥80	≥73 ‹80	≥66 ‹73	≥60 ‹66	≥55 ‹60	≥50 ‹55	≥45 ‹50	‹45
Letter Grade	AA	AB	BB	BC	CC	CD	DD	FF
Grade Point	10	9	8	7	6	5	4	0

8.2 The student's performance in any semester will be assessed by the Semester Grade Point Average (SGPA). Similarly, his performance at the end of two or more consecutive semesters will be denoted by the Cumulative Grade Point Average (CGPA). The SGPA and CGPA are calculated as follows:

(i) SGPA = $\Sigma C_i G_i / \Sigma C_i$ where C_i is the number of credits of course i
G_i is the Grade Point for the course i
and i = 1 to n, n = number of courses in the semester

(ii) CGPA = $\Sigma C_i G_i / \Sigma C_i$ where C_i is the number of credits of course i
G_i is the Grade Point for the course i
and i = 1 to n, n = number of courses of all semesters up to which CGPA is computed.

(iii) No student will be allowed to move further if CGPA is less than 3 at the end of every academic year.

(iv) A student will not be allowed to move to third year if he/she has not cleared all the courses of first year.

(v) A student will not be allowed to move to fourth year if he/she has not cleared all the courses of second year.

9. Awards of Degree

9.1 Every student of the programme who fulfils the following criteria will be eligible for the award of the degree:

9.1.1 He should have earned at least minimum required credits as prescribed in course structure; and

9.1.2 He should have cleared all internal and external evaluation components in every course; and

9.1.3 He should have secured a minimum CGPA of 4.5 at the end of the programme;

9.1.4 In addition to above, the student has to complete the required formalities as per the regulatory bodies, if any.

9.2 The student who fails to satisfy minimum requirement of CGPA at the end of program will be allowed to improve the grades so as to secure a minimum CGPA for award of degree. Only latest grade will be considered.

10. Award of Class

The class awarded to a student in the programme is decided by the final CGPA as per the following scheme:

Distinction:	CGPA ≥ 7.5
First class:	CGPA≥ 6.0
Second Class:	CGPA≥ 5.0

11. Transcript

The transcript issued to the student at the time of leaving the University will contain a consolidated record of all the courses taken, credits earned, grades obtained, SGPA, CGPA, class obtained, etc.

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY
(CHARUSAT)

FACULTY OF TECHNOLOGY & ENGINEERING

(FTE)

CHOICE BASED CREDIT SYSTEM

FOR

BACHELOR OF TECHNOLOGY & ENGINEERING

Choice Based Credit System

With the aim of incorporating the various guidelines initiated by the University Grants Commission (UGC) to bring equality, efficiency and excellence in the Higher Education System, Choice Based Credit System (CBCS) has been adopted. CBCS offers wide range of choices to students in all semesters to choose the courses based on their aptitude and career objectives. It accelerates the teaching-learning process and provides flexibility to students to opt for the courses of their choice and / or undergo additional courses to strengthen their Knowledge, Skills and Attitude.

1. CBCS – Conceptual Definitions / Key Terms (Terminologies)

1.1. Core Courses

1.1.1 University Core (UC)

University Core Courses are those courses which all students of the University of a Particular Level (PG/UG) will study irrespective of their Programme/specialisation.

1.1.2 Programme Core (PC)

A 'Core Course' is a course which acts as a fundamental or conceptual base for Chosen Specialisation of Engineering. It is mandatory for all students of a particular Programme and will not have any other choice for the same.

1.2 Elective Course (EC)

An 'Elective Course' is a course in which options / choices for course will be offered. It can either be for a Functional Course / Area or Streams of Specialization / Concentration which is / are offered or decided or declared by the University/Institute/Department (as the case may be) from time to time.

1.2.1 Institute Elective Course (IE)

Institute Courses are those courses which any students of the University/Institute of a Particular Level (PG/UG) will choose as offered or decided by the University/Institute from time-to-time irrespective of their Programme /Specialisation

1.2.2 Programme Elective Course (PE):

A 'Programme Elective Course' is a course for the specific programme in which students will opt for specific course(s) from the given set of functional course/ Area or Streams of Specialization options as offered or decided by the department from time-to-time

1.2.3 Cluster Elective Course (CE):

An 'Elective Course' is a course which students can choose from the given set of functional course/ Area or Streams of Specialization options (eg. Common Courses to EC/CE/IT/EE) as offered or decided by the Institute from time-to-time.

1.3 Non-Credit Course (NC) - AUDIT Course

A 'Non-Credit Course' is a course where students will receive Participation or Course Completion certificate. This will not be reflected in Student's Grade Sheet. Attendance and Course Assessment is compulsory for Non-Credit Courses.

Charotar University of Science & Technology

Chandubhai S Patel Institute of Technology

Devang Patel Institute of Advance Technology and Research

Department of Information Technology

To become a front-runner for quality education, development and research in the field of IT.

Mission

- *To prepare next-generation technocrats for societal upliftment.*
- *To inculcate moral and ethical values for building vibrant nation.*

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY (CHARUSAT)

TEACHING & EXAMINATION SCHEME FOR B TECH PROGRAMME IN IT

Sem	Course Code	Course Title	Teaching Scheme						Examination Scheme				
			Contact Hours				Credit		Theory		Practical/Project		Total
			Theory	Practical	Tutorial	Project	Theory	Practical	Internal	External	Internal	External	
SY Sem-3	MA253	Discrete Mathematics and Algebra	4	0	0	0	4	0	30	70	0	0	100
	IT259	Data Structures & Algorithms	3	2	0	0	3	1	30	70	25	25	150
	IT267	Java Programming	1	4	1	0	1	2	15	35	50	50	150
	XXXX	University Elective- I	2				2		100				100
	IT260	Database Management System	3	2	0	0	3	1	30	70	25	25	150
	HS121.02A	Creativity, Problem Solving and Innovation	0	2	0	0	0	2	0	0	30	70	100
	IT262	Web Technologies	0	4	0	0	0	2	0	0	50	50	100
			13	**14**	**1**	**0**	**13**	**10**	**205**	**245**	**180**	**220**	**850**
SY Sem-4	MA262	Statistical and Numerical Techniques	3	2	0	0	3	1	30	70	25	25	150
	IT261	Computer Networks	3	2	0	0	3	1	30	70	25	25	150
	IT263	Computer Architecture & Microprocessor Interfacing	4	2	0	0	4	1	30	70	25	25	150
	IT264	Full Stack Web Development	0	4	0	0	0	2	0	0	50	50	100
	IT265	Design and analysis of Algorithms	4	2	0	0	4	1	30	70	25	25	150
	IT266	Project – 1	0	0	0	2	0	1	0	0	25	25	50
	HS Elective	HS111.03 A Human Value and Ethics / HS113A Indian Knowledge System	0	2	0	0	0	2	0	0	30	70	100
	XXXXX	University Elective- II	2				2		0	0	30	70	100
			14	**16**	**0**	**2**	**14**	**9**	**220**	**280**	**265**	**385**	**1050**

B. Tech. (CE/CSE/IT/EC)Programme

B. Tech. (Information Technology) Programme

SYLLABI

(Semester – 3)

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

FACULTY OF SCIENCES
MATHEMATICAL SCIENCES
MA253: DISCRETE MATHEMATICS AND ALGEBRA

A. Credits and Hours:

Teaching Scheme	Theory	Practical	Total	Credit
Hours/week	4	-	4	4
Marks	100	-	100	

B. Outline of the course:

Sr No.	Title of the unit	Minimum number of hours
1.	Predicate Calculus	08
2.	Relations and Lattice	10
3.	Graph Theory	12
4.	Recurrence Relations	05
5.	Abstract Algebra	09
6.	Linear Algebra	16
	Total Hours	60

C. Detailed Syllabus:

1. Predicate Calculus: 08 Hours 13 %

1.1 Revision: Propositions, connectives, converse, inverse, contrapositive,tautology, contradiction.
1.2 Logical equivalence.
1.3 Minimal functionally complete set of connectives.
1.4 Principle conjunctive normal forms and Principle disjunctive normalforms.
1.5 Predicate calculus using rules of inferences.

2. Relations and Lattice: 10 Hours 17%

2.1 Revision of properties of relations on sets.
2.3 Representations of relations: graphical and matrix representation.
2.4 Equivalence relation, covering of a set, partition of a set.
2.5 Partially ordered sets, totally ordered sets, Hasse diagram.
2.6 Lattices, sub lattices.
2.7 Properties of lattices (without proof).
2.8 Complete lattices, bounded lattices, distributive lattices, complemented lattices and complemented distributive lattices.

3. Graph Theory: 12 Hours 20%

3.1 Basic terminologies, Simple graph, Types of graphs.
3.2 Degree of a vertex, matrix representations of graph.

3.3 Path and connectivity.
3.4 Eulerian and Hamiltonian graph.
3.5 Subgraphs, spanning subgraphs, isomorphic graphs.
3.6 Planar graphs.
3.7 Matching in graphs.
3.8 Graph coloring.

4. Recurrence Relations: 05 Hours 08%

4.1 Solutions of recurrence relation by direct methods.
4.2 Generating functions and solutions of recurrence relation.

5. Abstract Algebra: 09 Hours 15%

5.1 Groupoid, semi group, monoid, group.
5.2 Order of a group, order of an element, Lagrange's theorem.
5.3 Subgroup, cyclic subgroup, permutation group.

6. Linear Algebra: 16 Hours 27%

6.1 Vector space: definition and examples. Subspaces.
6.2 Linear combinations, linearly dependence and linearly independence.
6.3 Basis and dimension of a vector space.
6.4 Linear transformations. Null space and range of a linear transformation.Rank - nullity theorem. Isomorphisms.

D. Instructional Method and Pedagogy:

- At the starting of the course, the course delivery pattern, prerequisite of the subjectmust be discussed.
- Lectures may be conducted with the aid of multi-media projector, black board, OHPetc.
- Attendance is compulsory in lectures/laboratory which carries a 5% component ofthe overall evaluation.
- Minimum two internal tests/ unit tests must be conducted and average of two will beconsidered as a part of 15% overall evaluation.
- Assignments based on course content will be given to the students at the end of eachunit/topic and will be evaluated at regular interval. It carries a weightage of 5%.
- Two Quizzes (surprise tests)/ oral test / viva will be conducted which carries 5%component of the overall evaluation.

E. Student Learning Outcomes:

At the end of the course the students would be able to

CO1	Develop logical argument using truth table and rules of inferences in predicate calculus.
CO2	Relation and types of relations define on sets and utilize it to construct Hasse diagram andlattices on sets.
CO3	Graph and types of the graphs and identify the real-world phenomena in terms of graphtheory.
CO4	The concept of recurrence, generating functions and their applications in solvingrecurrence relations.
CO5	Different algebraic structures like groupoid, semi group, monoid, group, cyclic group andpermutation group

CO6	Definition of vector space, concepts of the terms: linear span, linear independence, basis, dimension. Definition and properties of linear transformations, range and kernel of a lineartransformation.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	-	1	-	-	-	-	-	-	-	2	-
CO2	3	2	-	-	-	-	-	-	-	-	-	-	3	-
CO3	3	2	1	-	2	-	-	-	-	-	-	-	3	1
CO4	3	-	-	-	1	-	-	-	-	-	-	-	3	-
CO5	3	1	-	-	-	-	-	-	-	-	-	-	2	-
CO6	2	1	-	1	-	-	-	-	-	-	-	-	2	-

Correlation levels 1, 2 or 3 as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

F. Recommended Study Material:

❖ Text Books:

1. Rosen, Kenneth H. and Kamala Krithivasan; Discrete mathematicsand its applications. Vol. 6. New York: McGraw-Hill, 1995.
2. Swapan Kumar Sarkar; A Text Book of Discrete Mathematics, S.Chand and Co. New Delhi 2008.
3. H. Anton and C. Rorres; Elementary Linear Algebra, Applicationversion, Wiley Edition 2010.

❖ Reference Books:

1. Jean-Paul Tremblay and Rampurkar Manohar; Discrete mathematical structures with applications to computer science. New York: McGraw-Hill, 1975.
2. D. F. McAllister and D. F. Stanat; Discrete Mathematics in Computer Science. Prentice-Hall, Inc.1977.
3. Narsingh Deo; Graph theory with applications to engineering and computer science. Courier Dover Publications, 2016.
4. B. Kolman and R. C. Busby; Discrete Mathematical Structures for ComputerScience, 2nd edition, Prentice-Hall, Englewood Cliffs, New Jersey 1987.
5. D. S. Malik and Mridul K. Sen; Discrete mathematical structures: theory andapplications. Course Technology, 2004.
6. H. Cormen Thomas, C. E. Leiserson, R. L. Rivest and C. Stein.; Introductionto algorithms (Vol. 6). Cambridge: MIT press,2001.

❖ URL Links:

Lecture Notes:

1. http://www.cs.yale.edu/homes/aspnes/classes/202/notes.pdf

2. http://home.iitk.ac.in/~arlal/book/mth202.pdf
3. https://web.stanford.edu/class/cs103x/cs103x-notes.pdf
4. https://www.cs.cornell.edu/~rafael/discmath.pdf
5. http://www-sop.inria.fr/members/Frederic.Havet/Cours/matching.pdf
6. http://wwwsop.inria.fr/members/Frederic.Havet/Cours/coloration.pdf

Video Lectures:

1. http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/video-lectures/

IT259: DATA STRUCTURES AND ALGORITHMS

A. Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	2	-	5	4
Marks	100	50	-	150	

Objective of the Course:

The main objectives for offering the course are:

- To familiarize students with basic data structures and their use in fundamental algorithms.
- To teach the students how to select and design data structures and algorithms for a specified problem.
- To teach the students how data will be stored efficiently within computer memory.
- To select appropriate data structure and algorithm for a specified application.

B. Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to Data Structure	04
2.	Linear Data Structure	12
3.	Non Linear Data Structure	16
4.	Searching and Sorting	10
5.	Hashing	03

Total hours (Theory) : 45
Total hours (Lab) : 30
Total hours : 75

C. Detailed Syllabus:

1. **INTRODUCTION TO DATA STRUCTURE** — **04 hours** — **9 %**

1.1 Introduction
Introduction to Data, Information, Data Type
Different types of Data Type : Built-In and Abstract Data Type

1.2 Algorithm and Data Structure
Algorithm, Program
Introduction to Data Structure, Needs for Data Structure Different types of Data Structure

2. **LINEAR DATA STRUCTURE** — **12 hours** — **27 %**

2.1 Array
Notations: one dimension, two dimension and multi dimension

	Memory Representation of Array: Row Order and Column Order Concept of Sparse Matrices		
2.2	Stack Memory Representation of Stack Operations: push, pop, peep, change Applications of Stack: Recursion: Recursive Function Tracing, Tower of Hanoi Conversion: Infix to Postfix Evaluation : Prefix and Postfix expression		
2.3	Queue Memory Representation of queue Simple Queue: Insert and Delete operation Circular Queue: Insert and Delete operation Concepts of: Priority Queue, Double-ended Queue Applications of Queue		
2.4	Linked List Memory Representation of LL Singly Linked List: Insert at First, Insert at End, Insert according to Sorted order, Delete the specified node. Doubly Linked List: Insert and Delete operation Concept of Circular Linked List Applications of Link List		
3.	**NON LINEAR DATA STRUCTURE**	**16 hours**	**35 %**
3.1	Tree Tree Concepts (Tree, Binary, Full Binary, Complete Binary) Memory Representation of Tree Tree Traversal Techniques: Pre-order, Post-order and In-order (Recursive and Iterative) Binary Search Tree: Iterative and Recursive: Insert and Delete Operations with all options. Concept of Threaded Binary Tree, B- Tree General Tree to Binary Tree Conversion Height-Balance Tree (AVL Tree): Insert and Delete Operations Applications of Tree : Manipulation of Arithmetic Expression, Decision Tree, Hierarchical Tree(Family Tree), Directory structure of File system		
3.2	Graph Graph concepts (undirected, directed, simple, multi, weighted, null, mixed, cycle, path, forest) Memory Representation of Graph, BFS and FS, Applications of Graph		
4.	**SEARCHING AND SORTING**	**10 hours**	**22 %**
4.1	Searching Sequential Search ,Binary Search : Iterative and Recursive		
4.2	Sorting Different Sorting Techniques Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Radix Sort, Heap Sort		
5.	**HASHING**	**03 hours**	**7 %**

5.1 Hashing
Collision-Resolution Techniques: rehashing and chaining
Different Hashing Functions: Division, Mid-square, Folding, Length-dependent, Digit Analysis, Multiplicative
Applications of Hashing

D. Instructional Method and Pedagogy:

At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.

• Lectures will be conducted with the aid of multi-media projector, black board, OHP etc. • Faculty would use coached problem solving method as it is class format in which faculty provide a structured, guided context for students working collaboratively to solve problems.

• Attendance is compulsory in lectures and laboratory which carries 5% component of the overall evaluation.

• Minimum two internal exams will be conducted and it will be considered as a part of 15% continuous evaluation.

• Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval. It carries a weight age of 5%.

• Surprise tests/Quizzes will be conducted which carries 5% component of the overall evaluation.

• The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.

• Minimum 10 experiments are suggested in the laboratory related to course content

E. Course Learning Outcomes

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

CO1	Describe how different data structures are represented in memory and used by algorithms/program.
CO2	Demonstrate different operations for various data structures
CO3	Describe and implement an appropriate data structure for various applications.
CO4	Apply and compare alternative implementations of different searching and sorting techniques with respect to performance

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	2	2	-	-	-	-	-	-	-	3	2
CO2	2	-	2	-	1	-	-	-	-	-	-	-	3	1
CO3	3	3	3	-	1	-	-	-	-	-	-	-	3	2
CO4	2	2	2	1	1	-	-	-	-	-	-	-	3	-

Enter correlation levels 1, 2 or 3 as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

If there is no correlation, put "-"

F. Recommended Study Material:

❖ **Text book:**

1. An Introduction to Data Structures with Applications, Jean-Paul Tremblay, Paul G. Sorenson, McGraw-hill.

❖ **Reference book:**

1. Classic Data structures, D.Samanta, Prentice-Hall International.
2. Data Structures using C & C++, Ten Baum, Prentice-Hall International.
3. Data Structures Using C, Oxford Higher Education, Reema Thareja
4. Data Structures: A Pseudo-code approach with C, Gilberg & Forouzan, Thomson Learning.
5. Fundamentals of Data Structures in C++, Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, W. H. Freeman.
6. Data Structure through C (A Practical Approach) , Dhanpat Rai & Co., G. S. Baluja

❖ **Web material:**

1. http://www.itl.nist.gov/div897/sqg/dads
2. http://www.leda-tutorial.org/en/official/ch02s02s03.html
3. http://www.leda-tutorial.org/en/official/ch02s02s03.html
4. http://www.softpanorama.org/Algorithms/sorting.shtml

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	2	0	5	4
Marks	100	50	0	150	

B. Outline of the Course:

Sr. No.	Title of theunit	Minimum number of hours
1.	Introductory concepts of DBMS	05
2.	Formal Relational Model and Query Languages	05
3.	Relational Database Design	10
4.	Transaction Management	10
5.	Concurrency Control and Recovery System	10
6.	Indexing	05
	Total hours (Theory):	45
	Total hours (Lab):	30
	Total hours:	75

C. Detailed Syllabus:

1. **Introductory concepts of DBMS** — **05 Hours** — **10%**

   Introduction and application of DBMS, Data Independence, Database System Architecture – levels, Mapping, Database users and DBA, Entity-Relationship model, constraints, keys, Design issues, E-R Diagram, Extended E-R features-Generalization, Specialization, Aggregation, Translating E-R model into Relational model.

2. **Formal Relational Model and Query Languages** — **05 Hours** — **12%**

   The relational Model, Structure of Relational Databases, Database Schema, Keys, Relational Algebra, Fundamental Relational Algebra operations, SQL fundamentals, DDL, DML, DCL Concepts, Cursors, Stored Procedures, Stored Functions, Database Integrity – Triggers.

3. **Relational Database design** — **10 Hours** — **22%**

   Functional Dependency–definition, Trivial and Non-Trivial FD, Closure of FD set, Closure of attributes, canonical cover, First, Second and Third Normal Forms, Dependency Preservation, Boyce-Codd Normal Form, Multi-valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

4.	**Transaction Management** Transaction concepts, A Simple Transaction Model, ACID properties, serializability of Transaction, Testing for Serializability.	**10 Hours**	**22%**
5.	**Concurrency Control and Recovery System** Lock based concurrency control, Deadlock, Two-phase locking protocol, Two- Phase Commit protocol, Time stamping methods, Database recovery management, Failure Classification, Recovery and Atomicity, Log-based recovery, Transaction rollback and checkpoints, System recovery.	**10 Hours**	**22%**
6.	**Indexing** Basic Concepts, Ordered Indices, B+ Tree Index Files, B-Tree Index Files.	**05 Hours**	**12%**

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, etc.
- Internal exams/Unit tests/Surprise tests/Quizzes/Seminar/Assignments etc. will be conducted as a part of continuous internal theory evaluation.
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Student Learning Outcome:

After learning the course, students will able to

CO1	Identify and evaluate the constructs in the E-R model and issues involved in developing an E-R diagram. Convert an E-R diagram into a relational database schema. Declare and enforce integrity constraints on database using a state-of- art RDBMS.
CO2	Demonstrate the basic elements of a relational database management system.
CO3	Design entity relationship, Convert entity relationship diagrams into RDBMS and formulate SQL queries on the data.
CO4	Demonstrate their understanding of transactions processing and recovery techniques to recover from the crashes.
CO5	Understand the uses of Database Schema and need of Normalization and Extend normalization for the development of application software 's.

Course Articulation Matrix:

	PO 01	PO 02	PO 03	PO 04	PO 05	PO 06	PO 07	PO 08	PO 09	PO 10	PO 11	PO 12	PSO 01	PSO 02
CO1	-	3	2	2	-	-	-	-	-	-	-	-	1	-
CO2	3	-	-	-	1	-	-	-	-	-	-	-	3	-
CO3	-	-	3	2	3	-	-	-	-	-	-	-	-	2
CO4	3	-	1	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	3	2	3	-	-	-	-	-	-	-	2	2

F. Recommended Study Material:

❖ Text Books:

- Database System Concepts, Abraham Silberschatz, Henry F. Korth & S.Sudarshan, McGraw Hill.
- An introduction to Database Systems, C J Date, Addition-Wesley

❖ Reference Books:

- "Fundamentals of Database Systems", R. Elmasri and S. B. Navathe, The Benjamin /Cumming Pub. Co
- SQL, PL/SQL the Programming Language of Oracle,IvanBayross, BPB PublicationsOracle: The Complete Reference, George Koch, Kevin Loney, Oracle Press.

❖ Web Materials:

- http://www.sql.org
- http://www.w3schools.com
- http://www.sqlcourse.com
- https://www.youtube.com/playlist?list=PLUd8M7XZdd6FT24ouEYl4RPpgXY9c_luI

IT262: WEB TECHNOLOGIES

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	0	4	0	4	2
Marks	0	100	0	100	

B. Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	HTML	6
2.	CSS	6
3.	JavaScript	10
4.	Cutting edge Frameworks for Web Development	8
	Total hours (Theory) :	00
	Total hours (Lab) :	30
	Total hours :	30

C. Detailed Syllabus:

1. **HTML** — **06 Hours** — **20%**

   HTML Styles, Links, Images, Tables, Lists, Forms, Frames, iframes,HTML semantic elements

2. **CSS** — **06 Hours** — **20%**

   CSS Introduction, CSS Syntax, CSS Id & Class, CSS Box Model, CSS Border, CSS Outline, CSS Margin, CSS Padding, CSS Styling (Backgrounds, Text, Fonts, Links, Lists, Tables), Pseudo-class, Pseudo-element, Navigation Bar, Image Gallery, Image Opacity, Image Sprites, Media Types, Attribute Selectors, CSS Text Effects, CSS 2D/3D Transformations, Transitions, CSS Animations, CSS Flex

3. **JavaScript** — **10 Hours** — **33%**

   Types of Data in JavaScript: Numerical Data, Text Data, Boolean Data, And Variables: Creating Variables and Giving Them Values, Assigning Variables with the Value of Other Variables. Data Type Conversion: Dealing with Strings That Won't Convert Decisions,Loops, and Functions: if...else, for loop for...in Loop, switch...case,
   while Loop, do...while loop, break and continue Statements, Creating

function with and without arguments, HTML Form and Validation: Button Elements, Text Elements, The textarea Element, Check Boxes and Radio Buttons, Selection Boxes, validation, JavaScript object: String, Array, Regular Expression

4. **Cutting edge framework for Web Development** **08 Hours** **27%**

JQuery, AJAX – front-end development,

BootStrap – CSS framework

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will bediscussed.
- Labs will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in laboratory.
- Assignments/ Surprise tests/Quizzes/Seminar based on course content will be given tothe students at the end of each unit/topic and will be evaluated at regular interval
- The course includes a laboratory, where students have an opportunity to build anappreciation for the concepts being taught.
- Experiments/Tutorials related to course content will be carried out in the laboratory.
- Students need to develop a project on technologies learned in laboratory sessions.

E. Student Learning Outcome:

After learning the course, students will able to

CO1	Student should able to understand the tools and technologies to design & develop static and dynamic webpages/apps
CO2	Student should able to build creative UI design for responsive/device independent webpages
CO3	Student should able to select appropriate hosting environment
CO4	Student should able to understand and apply concepts of web security through session and cookies
CO5	Student should able to design, develop and deploy multi-tier web applications

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12	PSO 01	PSO 02
CO1	1	2	3	-	-	-	-	-	-	-	-	1	1	-
CO2	-	-	-	2	3	-	1	-	1	2	-	1	2	-

CO3	-	2	-	1	2	-	-	-	-	-	1	-	-	1
CO4	3	-	2	-	-	3	1	2	-	-	1	-	3	2
CO5	3	1	3	-	3	2	-	-	1	-	-	1	2	-
CO6	-	-	-	-	-	-	-	-	-	-	-	-	-	-

F. **Recommended Study Material:**

- **Text Books:**
  - HTML5 and CSS3, Second Edition Level Up with Today's Web Technologies, Brian P. Hogan

- **Reference Books:**
  - HTML & CSS, Design and Build Websites, Jon Duckett

- **Web Material:**
  - https://w3schools.com
  - http://angularcasts.io/

IT267: JAVA PROGRAMMING

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	1	4	1	6	3
Marks	50	100	0	150	

B. Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Fundamental of Object-Oriented Programming	2
2.	Class Fundamentals	4
3.	Array & String Handling	4
4.	Inheritance, Interfaces & Packages	5
5.	Exceptions Handling	5
6.	Multithreaded Programming	4
7.	GUI Programming & Lambdas and Streams	4
8.	File I/O and NIO	2
	Total hours (Theory) :	30
	Total hours (Lab) :	60
	Total hours :	105

C. Detailed Syllabus:

1. **Fundamental of Object-Oriented Programming** **02 Hours** **04 %**

   History of Java, Basic overview of java, Bytecode, JVM, Buzz-words, Application and applets, Constants, Variables & Data Types, Comments, Operators, Control Flow

2. **Class Fundamentals** **04 Hours** **09 %**

   General form of class, Creating class Overloading methods, Constructor, Declaring Object, Returning objects, using objects as parameters, Assigning object reference variables, Introducing Access control, Understanding static, Introducing final, The finalize() method, The this keyword, Garbage collection

3.	**Array & String Handling** Array basics, String Array, String class, StringBuffer class, String Tokenizer Class and Object Class	**04 Hours**	**04%**
4.	**Inheritance, Interfaces & Packages** Inheritance: Using super creating multilevel Hierarchy, method overriding, Dynamic method dispatch, abstract classes, using final with Inheritance, Using Package: Defining package, finding package and CLASSPATH, Access protection, importing package, Interface: Defining Interface, Implementing Interface, Variables in Interface	**05 Hours**	**13 %**
5.	**Exceptions Handling** Exception types, Try ...Catch...Finally, Throw, Throws, creating your own exception subclasses	**05 Hours**	**11 %**
6.	**Multithreaded Programming** Life cycle of thread, thread methods, thread priority, thread exceptions, Implementing Runnable interface, Synchronization	**04 Hours**	**16 %**
7.	**GUI Programming & Lambdas and Streams** Introduction to Annotation, Byte streams and character streams, Wrapper classes , Why Lambda Expression, Lambda Expression Syntax, Where to use lambda expression, Adopting Patterns like matching, finding and filtering, Swing overview ,Swing component classes: AbstractButton, ButtonGroup, ImageIcon, JApplet, Jbutton, JcheckBox, JcomboBox, Jlabel, JradioButton, JscrollPane, JtabbedPane, Jtable, JtextField, Jtree	**04 Hours**	**16 %**
8.	**Java I/O** File and Directories, Byte streams and character streams, Random Access Files	**02 Hours**	**13 %**

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHPetc.
- Attendance is compulsory in lectures and laboratory which carries 10 Marks weight.
- Assignments/ Surprise tests/Quizzes/Seminar based on course content will be given tothe students at the end of each unit/topic and will be evaluated at regular interval
- The course includes a laboratory, where students have an opportunity to build anappreciation for the concepts being taught in lectures.

- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Student Learning Outcome:

After learning the course, students will able to

CO1	Implement Object Oriented programming concept using basic syntaxes of control Structures, strings, and function for developing skills of logic building activity.
CO2	Use of a variety of basic control structures including selection and repetition;classes and objects in a tiered architecture (user interface, controller, and application logic layers)
CO3	Demonstrates how to achieve reusability using inheritance, interfaces, and packages and describes faster application development that can be achieved.
CO4	Demonstrate understanding and use of different exception handling mechanismsand concepts of multithreading for robust faster and efficient application development.
CO5	Identify and describe common abstract user interface components to design GUI in Java using Swing along with a response to events.
CO6	Identify, Design & develop complex Graphical user interfaces using principal Java Swing classes based on MVC architecture

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO 05	PO 06	PO 07	PO 08	PO 09	PO 10	PO11	PO12	PSO 01	PSO 02
CO1	2	3	3	-	-	-	-	-	-	-	-	-	-	-
CO2	3	2	3	2	1	1	-	-	-	-	-	-	1	3
CO3	-	2	2	3	2	-	-	-	-	-	1	1	-	-
CO4	1	1	1	1	3	-	-	-	-	-	2	2	3	-
CO5	1	1	2	3	3	-	-	1	-	-	1	1	2	2
CO6	1	1	1	2	3	1	-	1	-	-	2	1	3	3

F. Recommended Study Material:

❖ Text Books:

1. Java: The Complete Reference, Eleventh Edition by Herbert Schildt, Oracle Press

❖ Reference Books:

1. Java: A Beginner's Guide, Eighth Edition 8th Edition by Herbert Schildt, OraclePress
2. Head First Java: A Brain-Friendly Guide 2nd Edition by Kathy Sierra, Bert Bates, O'Reilly
3. OCP Oracle Certified Professional Java SE 11 Programmer I Study Guide: Exam 1Z0-815 1st Edition by Jeanne Boyarsky, Scott Selikoff

❖ Web Materials:

1. https://docs.oracle.com/javase/tutorial/tutorialLearningPaths.html

2. http://openjdk.java.net/projects/jigsaw/
3. https://docs.oracle.com/en/java/javase/14/docs/api/index.html

❖ **Software**

1. https://www.oracle.com/java/technologies/javase-downloads.html
2. https://netbeans.apache.org/download/index.html
3. https://download.eclipse.org/eclipse/downloads/

SYLLABI

(Semester – 4)

A. Credits and Hours:

Teaching Scheme	Theory	Practical	Total	Credit
Hours/week	3	2	5	4
Marks	100	50	150	

B. Outline of the course:

Sr No.	Title of the unit	Minimum number of hours
1.	Sampling Distributions and Test of Hypotheses	12
2.	Simulation	12
3.	Simple Regression and Simple Correlation	06
4.	Interpolation and Curve Fitting	15
5.	Numerical Integration, Solution of Different Typesof Equations.	15
	Total hours	60

C. Detailed Syllabus:

1. Sampling Distributions and Test of Hypotheses: 12 Hours 20%

1.1 Population and sample, function of random variables associated with normal Distribution, Central limit theorem.

1.2 Random sampling, Sample moments and their distributions: Chi-square, t and F distributions.

1.3 Point estimation and interval estimation: Estimation of population mean, population variance, population proportion, one population and two populations.

1.4 Introduction to hypothesis Testing, z- test, t-test, chi-square test and F-test, one sample and two samples tests.

2. Simulation: 12 Hours 20%

2.1 Introduction to random numbers.

2.2 Generating random numbers from probability distributions: Binomial, Poisson, Uniform, Exponential and Normal.

2.3 Variance reduction techniques.

2.4 Markov Chain, Monte Carlo Method and its applications.

3. Simple Regression and Simple Correlation: 06 Hours 10%

3.1 Measure of association between two variables. Types of correlation, Karl Pearson's Coefficient of correlation and its mathematical properties.

3.2 Spearman's Rank correlation and its interpretations.

3.3 Regression Analysis: Concept and difference between correlation and regression, linear regression equations, properties of regression coefficients.

4. Interpolation and Curve fitting: 15 Hours 25%

4.1 Errors in numerical analysis: types of errors, sources of errors.

4.2 Interpolation, Lagrange's interpolation formula. Newton's divided difference table and Newton's Interpolation polynomial.

4.3 Finite differences and associated operators.

4.4 Newton's forward interpolation formula, Newton's backward interpolation formula.

4.5 Least squares curve fitting methods, linear and quadratic curve fitting.

5. Numerical Integration and Numerical Solution of Different Types of Equations: 15 Hours 25%

5.1 Numerical Integration: Rectangle rule, trapezoidal rule and Simpson's rules (1/3 and 3/8) and their composite rules.

5.2 Numerical solution of equations: Bisection method, False position (Regula-Falsi) and Newton-Raphson method.

5.3 Numerical solution of system of simultaneous linear equations: Gauss Jacobi Method and Gauss Seidel Method.

5.4 Numerical Solution of Ordinary Differential Equations: Taylor's series, Euler's, and Runge- Kutta (2^{nd} and 4^{th} order) methods.

D. Instructional Method and Pedagogy:

- At the starting of the course, the course delivery pattern, prerequisite of the subject tobe discussed.
- Lectures may be conducted with the aid of multi-media projector, black board, OHPetc.
- Attendance is compulsory in lectures which carries a 5% component of the overallevaluation.
- Minimum two internal tests/unit test must be conducted and average of two will beconsidered as a part of 15% overall evaluation.
- Quizzes (surprise tests) /Oral tests/ Viva/Assignments will be conducted whichcarries 10% component of the overall evaluation.

E. Student Learning Outcomes:

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

CO1.	• Differentiate between population and sample distribution. Parameter andStatistic • Calculate confidence interval for parameter • Formulate null and alternate hypothesis • Solve the test of hypothesis problems

CO2.	• Relate the real system with virtual system • Construct simulation algorithm to generate random numbers • Verify the stated results of variance reduction
CO3.	• Identify the nature of relationship between two variables • Describe the scatter diagrams • Calculate correlation coefficient, regression coefficients
CO4.	• Examine the types of error in numerical computations • Differentiate the interpolation techniques and curve fitting techniques • Create the divided difference table • Solve the problems of interpolation • Solve the problems of curve fitting
CO5.	• Identify the types of equations • Solve the problems of numerical integration • Reproduce the algorithms of numerical solution of equations • Calculate the errors of approximations.
CO6.	• Follow the various techniques of statistical methods and numerical methods • Adopt the applications of these methods using computer • Create the computer algorithms of these methods

Course Articulation Matrix:

	PO 01	PO 02	PO 03	PO 04	PO 05	PO 06	PO 07	PO 08	PO 09	PO 10	PO 11	PO 12	PSO 01	PSO 02
CO1	3	3	-	1	1	-	-	-	-	-	-	-	3	-
CO2	3	3	-	3	3	-	-	-	-	-	-	-	3	-
CO3	3	3	-	1	1	-	-	-	-	-	-	-	3	-
CO4	3	3	-	1	1	-	-	-	-	-	-	-	3	-
CO5	3	3	-	1	1	-	-	-	-	-	-	-	3	-
CO6	3	3	-	3	3	-	-	-	-	-	-	-	3	-

- Correlation levels 1, 2 or 3 as defined below:
- 1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

F. Recommended Study Material:

❖ Text Books:

1. Richard A. Johnson, Miller and Freund; Probability and Statistics for Engineers.Prentice Hall, 1994.
2. Ross Sheldon; A first course in probability. Pearson Education India, 2002.
3. Ross Sheldon; A course in simulation. Prentice Hall PTR, 1990.
4. Shankar S. Sastry; Introductory methods of numerical analysis. PHI Learning Pvt.Ltd., 2015.

❖ Reference Books:

1. Robert V. Hogg, Elliot Tanis and Dale Zimmerman; Probability and

statisticalinference. Pearson Higher Ed, 2014.

2. Kishor S. Trivedi; Probability and statistics with reliability, queuing and computerscience applications. John Wiley & Sons, 2008.
3. Steven C.Chapra and Raymond P. Canale; Numerical methods for engineers. Vol.
   2. New York: McGraw-Hill, 2012.
4. VaidyeswaranRajaraman; Computer oriented numerical methods. PHI Learning Pvt.Ltd., 1993.
5. Erwin Kreyszig: Advanced Engineering Mathematics, 9th Ed., Jhon Wiley & Sons,India, 1999.

❖ **URL Links:**

1. http://numericalmethods.eng.usf.edu
2. http://mathworld.wolfram.com/
3. http://en.wikipedia.org/wiki/Math

A. Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	2	0	5	4
Marks	100	50	0	150	

B. Outline of the Course:

Sr. No.	Title of the Unit	Minimum number of hours
1.	Computer Networks and the Internet	03
2.	Application Layer	10
3.	Transport Layer	13
4.	The Network Layer	10
5.	The Link Layer: Links, Access Networks, and LANs	06
6.	Network Management	03
	Total hours (Theory)	45
	Total hours (Lab)	30
	Total hours	75

C. Detailed Syllabus:

1.	**Computer Networks and the Internet**	03 hours	08 %
1.1	What Is a Protocol?		
1.2	Access Networks		
1.3	Physical Media		
1.4	Packet Switching & Circuit Switching		
1.5	Delay, Loss, and Throughput in Packet-Switched Networks		
2.	**Application Layer**	10 hours	22 %
2.1	Principles of Network Applications		
2.2	The Web and HTTP		
2.3	File Transfer: FTP		
2.4	SMTP		
3.	**Transport Layer**	13 hours	30 %
3.1	Introduction and Transport-Layer Services		
3.2	Multiplexing and DE multiplexing		
3.3	Connectionless Transport: UDP		
3.4	Principles of Reliable Data Transfer		

3.5	Connection-Oriented Transport: TCP		
3.6	Principles of Congestion Control		
4.	**The Network Layer**	**10 hours**	**22 %**
4.1	Introduction		
4.2	Virtual Circuit and Datagram Networks		
4.3	What's Inside a Router?		
4.4	The Internet Protocol (IP): Forwarding and Addressing in the Internet		
4.5	Routing Algorithms		
5.	**The Link Layer: Links, Access Networks, and LANs**	**06 hours**	**12 %**
5.1	Introduction to the Link Layer		
5.2	Error-Detection and -Correction Techniques		
5.3	Multiple Access Links and Protocols		
5.4	Switched Local Area Networks		
6	**Network Management**	**03 hours**	**07 %**
6.1	What Is Network Management?		
6.2	The Infrastructure for Network Management		
6.3	The Internet-Standard Management Framework		
7	**Self-Study Topics**		
	Data Centre Networking, Socket Programming with UDP, Socket Programming with TCP		

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lectures and laboratory which carries 5 Marks weightage.
- Two internal exams will be conducted and average of the same will be converted to equivalent of 15 Marks as a part of internal theory evaluation.
- Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval. It carries a weightage of 5 Marks as a part of internal theory evaluation.
- Surprise tests/Quizzes/Seminar will be conducted which carries 5 Marks as a part of internal theory evaluation.
- The course includes a laboratory, where students have an opportunity to build anappreciation for the concepts being taught in lectures.

- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Student Learning Outcome:

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

CO1	Analyze layered network architecture and passage of data over communicationlinks
CO2	Analyze delay models in Data Networks using Queueing Systems for messaging and delay sensitive applications
CO3	Design and analyze routing algorithms for Internet and multi-hop autonomousnetworks
CO4	Analyze flow and rate control algorithms between a sender and receiver in widearea networks
CO5	Apply the network fundamentals to analyze performance.
CO6	Use key networking algorithms in simulation.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	3	1	-	-	-	-	-	-	-	2	-
CO2	3	3	1	3	1	-	-	-	-	-	-	-	1	-
CO3	3	3	1	3	1	-	-	-	-	-	-	-	1	-
CO4	3	3	1	3	1	-	-	-	-	-	-	-	1	-
CO5	3	-	-	-	-	-	-	-	-	-	-	-	-	-
CO6	-	-	-	-	3	-	-	-	-	-	-	-	-	-

F. Recommended Study Material:

❖ Text Book

1. Computer Networking: A Top-Down Approach James F. Kurose, University of Massachusetts, Amherst Keith W. Ross, Polytechnic University, Brooklyn

❖ Reference Materials:

1. Computer Networks by Andrew S Tanenbaum.
2. Data Communication And Networking by Behrouz Forouzan

❖ Web Materials:

1. www.ietf.org – For drafts
2. www.ieee.org – For standards and technical research papers
3. http://nptel.iitm.ac.in/courses.php?disciplineId=117

IT263: COMPUTER ARCHITECTURE & MICROPROCESSOR INTERFACING

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	4	2	-	6	5
Marks	100	50	-	150	

Objective of the Course:

The main objectives for offering the course are:

- To provide introduction to Instruction Set Architecture and Practical exposure through simulation tools/Microprocessor Kits
- To explore the basic concepts of computer organization & computer architecture design, Computer System Components: Processor, Memory, and I/O Devices, Performance evaluation
- To provide insight details in Processor Components: Control Unit, Registers, Caches Memory, ALU, and Instruction Execution Unit.

B. Outline of the Course:

Total hours (Theory): 60

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to Computer Architecture	03
2.	Instruction Set Architecture	05
3.	Computer Architecture Space	06
4.	Performance Measures	04
5.	Basics of Arithmetic Logic Unit	10
6.	Processor Design	10
7.	Pipelined Processor	06
8.	Memory Hierarchy	08
9.	Input/Output Subsystem	08

Total hours (Lab): 30

Total hours: 95

C. Detailed Syllabus:

1. **Introduction to Computing Systems** — **03 Hours** **5%**

   What is Computer Architecture, Abstraction: Software & Hardware ,

   Architecture Levels, Embedded Computers, Different types of processors, Five generation computers

   Looking into future: Grid Computing, Nano Computing, DNA Computing, Quantum Computing

2. **Instruction Set Architecture** — **05 Hours** **8%**

   Instruction for arithmetic, Instructions to move data, Instruction for decision making, Handling Constant Operands, Implementing loops, pointers Vs Index, Switch

Statement, Addresses in MIPS Instructions, Procedural abstractions, Requirements, Sorting example, Register use conventions, Recursive Programs: Activation Record, Calls, Returns(after instruction set architecture)

3.	**Computer Architecture Space** Architecture Space: MIPS ISA Features, Alternative Architectures Architecture Examples: RISC and CISC, PowerPC, VAX, SPARC, Intel x86	06 Hours	10%
4.	**Performance Measures** Performance and Cost, Purchasing perspective, Design perspective Notions of Performance: Latency and throughput, Performance and time, computer clocks, Computing CPU time and cycles, Improving Performance, Linking instruction, cycles and time, CIPS and MIPS examples, Computer Benchmarks, Sources of Benchmark: SPEC 89 and SPEC 95. Amdahl's law, Estimating performance improvements, poor performance metrics	04 Hours	7%
5.	**Basics of Arithmetic Logic Unit** Binary Arithmetic, ALU Design, Signed Operations and Overflow, Multiplier Design, Divider Design, Fast Addition, Multiplication, Floating Point representation and operations, Floating Point Unit Design, Floating Point Arithmetic	10 Hours	17%
6.	**Processor Design** Introduction, Simple Design Multi cycle approach, control for multi cycle, Micro-programmed Control, Exception Handling	10 Hours	17%
7.	**Pipelined Processor** Basic Design Idea, Data path and Control, Handling Data Hazards, handling Control Hazards	06 Hours	10%
8.	**Memory Hierarchy** Basic Idea: Memory construction, size, speed, cost and data unit. Tradeoffs between them. PROM, EEPROM, DRAM, SRAM, Memory Technologies, Hierarchical organization, principle of locality, Simple Cache organization, Miss rate, block size, cache policies Cache Organization: Mapping alternatives- direct, associative and set associative, processor performance with cache, memory organization and miss penalty, Policies for read, load, fetch, replacement and write, How Caches work, Size of tags, Performance analysis examples Virtual Memory: Similarities and differences of Virtual Memory and Cache, Mapping Virtual address to physical address, Page tables, TLB, Virtually addressed cache, Memory Protection	08 Hours	13%
9.	**Input/output Subsystem** Interfaces and buses, I/O operations, Designing I/O systems	08 Hours	13%

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.

- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lectures and laboratory which carries 10 Marks weight.
- Assignments/ Surprise tests/Quizzes/Seminar based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Student Learning Outcome:

After learning the course, students will able to

CO1	Identify the addressing mode of instructions and write machine program
CO2	Determine which hardware blocks and control lines are used for specific instructions
CO3	Demonstrate how to add and multiply integers and floating-point numbers using two's complement and IEEE floating point representation
CO4	Use various metrics to calculate and Analyze clock periods, performance, and instruction throughput of single-cycle, multi-cycle, and pipelined implementations of a simple instruction set
CO5	Detect pipeline hazards and identify possible solutions to those hazards to take advantage of super scalar architecture
CO6	Show how cache design parameters affect the performance of program and Map a virtual address into a physical address

Course Articulation Matrix:

	PO 01	PO 02	PO 03	PO 04	PO 05	PO 06	PO 07	PO 08	PO 09	PO 10	PO 11	PO 12	PSO 01	PSO 02
CO1	1	-	3	-	2	-	-	-	-	-	-	-	2	-
CO2	1	1	2	-	-	-	-	-	-	-	-	-	1	-
CO3	1	-	-	-	-	-	-	-	-	-	-	-	-	2
CO4	2	3	-	3	3	-	-	-	-	3	-	-	2	-
CO5	1	2	1	1	-	-	-	-	-	-	-	1	-	2
CO6	1	1	-	-	-	-	-	-	-	-	-	1	-	1

F. Recommended Study Material:

❖ Text Books:

1. John L. Hennesy & David A. Patterson, Computer Organization and Design MIPS Edition: The Hardware/Software Interface (The Morgan Kaufmann Series in Computer Architecture and Design)

❖ Reference Books:

1. R. E. Bryant and D. R. O'Hallaron, Computer Systems: A Programmer's Perspective, Prentice Hall
2. Computer Organization & Architecture-Designing for Performance, William Stalling, Pearson Prentice Hall (8th Edition).

3. Introduction to Computing Systems: From Bits and Gates to C and Beyond, Yale N. Patt,Sanjay J. Patel, 2nd Edition, Tata McGraw-Hill Publication,2005.
4. Structured Computer Organization, A. S. Tananbum , Pearson Education
5. The Essentials of Computer Organization And Architecture, Linda Null, Julia Lobur, Jones & Bartlett Learning, 2006
6. Computer Architecture & Organization, John P Hayes, McGraw-Hill.
7. Computer System Architecture, Morris Mano (3rd Edition) Prentice Hall.

❖ **Web Materials:**

1. http://pages.cs.wisc.edu/~markhill/cs354/Fall2008/notes/flpt.apprec.html
2. https://www.youtube.com/watch?v=qlH4-oHnBb8
3. https://nptel.ac.in/courses/106105033/ (For cache memory and Pipelining)
4. https://www.ebookbou.edu.bd/Books/Text/SST/DCSA/dcsa_2301/Unit-08.pdf
5. https://www.youtube.com/playlist?list=PLxCzCOWd7aiHMonh3G6QNKq53C6oNXGrX

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	0	4	-	4	2
Marks	0	100	-	100	

Objective of the Course:

The main objectives for offering the course are:

- To use React JS for front-end development
- To use Node JS, Express JS for back-end development
- To explore various deployment environments
- To use various testing tools and version control.

B. Outline of the Course:

Sr. No.	Title of the Unit	Minimum Number of Hours
1.	Front end Development	20
2.	Back-end Development	20
3.	NoSQL Database	10
4.	Version Control & Code Analysis	04
5.	Testing & Deployment	06

Total hours (Theory): 00

Total hours (Lab): 60

Total hours: 60

C. Detailed Syllabus:

1. Front end Development – React JS — 10 Hours — 33.33 %

   JavaScript Refresher

   Understanding JSX

   React Components & Hooks

   Styling Components

   Working with Environment Variables

   Understanding Props

   Understanding State & Handling Events in React

   Working with Forms

   Adding Router to the Application

   Refactoring App Component

	Hosting React App on Cloud Environments API Integration & Testing		
2.	Back-end Development – Node JS, Express JS Basics of Node JS & Execution Environment & Flow Setting Up Node JS Environment & Node Version Management Working with Node REPL Module Fundaments Exploring, Accessing & Building Modules Introduction to Back-End Frameworks Introduction and Setting Up / Configuring Express JS Understanding Express JS Project Structure & Scaffolding Understanding Middleware's & Routes Working with Environment Variables Database Integration Accessing & Building REST APIs Securing REST APIs Testing API	10 Hours	33.33 %
3.	NoSQL Database – MongoDB Introduction to NoSQL Setting Up / On Premise / Cloud Environments Working with shell and Server Basic Syntax, Schemas and Relation Basics and CRUD Operation Working with indexes & Optimization Aggregation Framework Security Basics	05 Hours	16.66 %
4.	Version Control & Code Analysis Code Version Management Code Structure and Organization Coding Standards and Best Practices Error Handling and Logging Code Performance and Optimization Refactoring Code	02 hours	06.68%
5.	Testing & Deployment Manual Testing Unit Testing	03 hours	10%

Hosting Selection Trade Offs

Preparing & Configuring Deployment Environment

App Deployment & Accessing

End to End Testing

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board etc.
- Attendance is compulsory in lectures and laboratory.
- Marks will be given based on continues evaluation, i.e. Unit Tests/Surprise tests/Quizzes/Projects/Presentation and Assignments based on course content will be given to the students at the end of each unit/topic.
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Course Outcome (COs):

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

CO1	Understanding Application Architecture and Environments
CO2	Developing cross-platforms Front End UI and Testing
CO3	Understanding Back-End and Developing REST APIs
CO4	Integrating Back-End with NoSQL Database & Applying Security
CO5	Developing Full stack App & Exploring Industry Best Practices
CO6	Configuring and Setting Up Cloud Environments & Deploying App

Course Articulation Matrix:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	-	-	-	2	-	-	-	-	-	-	-	-
CO2	-	3	-	-	-	2	-	-	-	3	-	-	-	-
CO3	-	-	3	-	-	-	-	3	-	3	-	-	-	-
CO4	1	-	-	3	-	2	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	3	-	-	-	-
CO6	-	-	-	-	2	2	3	-	-	3	-	-	-	-

Enter correlation levels 1, 2 or 3 as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

If there is no correlation, put "-"

F. Recommended Study Material:

❖ **Reference Links/ e-content:**

1. https://reactjs.org/docs/getting-started.html
2. https://nodejs.org/en/docs/guides/
3. https://expressjs.com/
4. https://www.mongodb.com/docs/
5. https://kafka.apache.org/documentation/
6. https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome.html

A. Credit and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	4	2	0	6	5
Marks	100	50	0	150	

B. Outline of the course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Basics of Algorithm and Mathematics	04
2.	Analysis of Algorithm	10
3.	Divide and Conquer Algorithm	08
4.	Greedy Algorithm	12
5.	Dynamic Programming	10
6.	Exploring Graphs	08
7.	String Matching and NP Completeness	08
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	75

C. Detailed Syllabus:

1. Basics of Algorithm and Mathematics **04 hours** **08%**

1.1 What is an algorithm?

1.2 Mathematics for Algorithm

1.3 Performance Analysis, Model for Analysis - Random Access Machine (RAM), Primitive Operations

1.4 Time Complexity and Space Complexity

2. Analysis of Algorithm **10 hours** **18%**

2.1 The efficiency of algorithm, Best, Average and Worst case Analysis

2.2 Asymptotic Notation

2.3 Solving Recurrence Equation

2.4 Sorting Algorithm

3. Divide and Conquer Algorithm **08 hours** **18%**

3.1 Basic of Recursion and its complexity

3.1	The general template for Divide and Conquer Problem		
3.2	Problem solving using divide and conquer algorithm – BinarySearch, Sorting - Merge Sort and Quick Sort		
3.3	Strassen's Matrix Multiplication		
4.	**Greedy Algorithm**	**10 hours**	**16%**
4.1	General Characteristics of greedy algorithms		
4.2	Problem solving using Greedy Algorithm: Making changeproblem The Knapsack Problem, Job Scheduling Problem		
4.3	Minimum Spanning Trees (Kruskal's Algorithm, Prim's Algorithm)		
4.4	Dijkstra Algorithm		
4.5	Huffman Coding Algorithm		
5.	**Dynamic Programming**	**12 hours**	**23%**
5.1	Introduction, The Principle of Optimality		
5.2	Problem Solving using Dynamic Programming – Calculating theBinomial Coefficient		
5.3	Making Change Problem, Assembly Line Scheduling		
5.4	Knapsack Problem, All pair Shortest Path		
5.5	Matrix Chain Multiplication		
5.6	Longest Common Subsequence		
5.7	Longest Increasing Subsequence		
6.	**Exploring Graphs and Backtracking**	**08 hours**	**08%**
6.1	An introduction to Graph, Basic Definitions		
6.2	Traversing Graphs – Depth First Search, Breadth First Search, Topological Sort		
6.3	Graph Representations – Adjacency Matrix, Adjacency List, Incidence Matrix		
6.4	Graph Applications – Shortest Path, Minimum Spanning Tree		
6.5	Backtracking – The Eight Queen Problem		
6.6	The Knapsack Problem		
6.7	Branch and Bound – The Assignment Problem		
6.8	Branch and Bound – Traveling Salesman Problem		
7.	**String Matching and NP Completeness**	**08 hours**	**08%**
7.1	Introduction		
7.2	The naïve string matching algorithm		

7.3 The Rabin-Karp algorithm

7.4 The Knuth-Morris-Pratt (KMP) algorithm

7.5 The Boyer-Moore algorithm

7.6 Introduction to NP Complete Theory

7.7 NP Completeness Proofs and Examples

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lectures and laboratory which carries 5 Marks weightage.
- Two internal exams will be conducted and average of the same will be converted to equivalent of 15 Marks as a part of internal theory evaluation.
- Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval. It carries a weightage of 5 Marks as a part of internal theory evaluation.
- Surprise tests/Quizzes/Seminar will be conducted which carries 5 Marks as a part of internal theory evaluation.
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments/Tutorials related to course content will be carried out in the laboratory.

E. Student Learning Outcome:

After completion of the course students will be able to

CO1	Students will able to analyze the performance of algorithms.
CO2	Students will able to select appropriate design techniques for effective solution of the problem.
CO3	Ability to find time and space complexity of the algorithm.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	1	1	-	-	-	-	-	-	-
CO2	3	3	3	3	2	-	-	-	-	-	-	-
CO3	3	3	1	2	1	-	-	-	-	-	-	-

Enter correlation levels 1, 2 or 3 as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial

(High)If there is no correlation, put "-"

F. Recommended Study Material:

❖ Text Books:

1. Gills Brassard, Paul Brately, Fundamental of Algorithms, Prentice Hall of India

❖ Reference Books:

1. Thomas H. Coreman, Charles E. Leiserson, Ronald Rivest and Clifford Stein, Introduction to Algorithms, MIT Press
2. Ellis Horowitz, Sartaz Sahni and Sanguthevar Rajasekarn Fundamental of ComputerAlgorithms, Computer Science Press

❖ URL Links:

1. http://www.itl.nist.gov/div897/sqg/dads
2. http://www.stanford.edu/class/cs161
3. http://highered.mcgraw-hill.com/sites/0073523402

IT266: PROJECT - I

A. Credits and Hours:

Teaching Scheme	Theory	Project	Tutorial	Total	Credit
Hours/week	0	2	0	2	1
Marks	0	50	0	50	

B. Outline of the Course:

- Students at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students are required to get approval of project definition from the department.
- After approval of project definition students are required to report their project work on weekly basis to the respective internal guide.
- Project will be evaluated at least once per week in laboratory hours during the semester and final submission will be taken at the end of the semester as a part ofcontinuous evaluation.
- Project work should include whole SDLC of development of software / hardware system as a solution of particular problem by applying principles of Software Engineering.
- Students have to submit project with following listed documents at the time of final submission.

  a Project Synopsis

  b. Software Requirement Specification

  c. SPMP

  d. Final Project Report/paper

  e. Project Setup file with Source code [Uploaded on GitHub]

  f. Project Presentation (PPT)

  g. Video Recording (Per Project)
- A student has to produce some useful outcome by conducting experiments orproject work.

Total hours (Theory): 60
Total hours (Lab): 30
Total hours: 90

C. Instructional Method and Pedagogy:

- Project Groups would be form of maximum two students.

- ❖ Inter batch group formation is not permitted due to difficulties in progress tracking.
- ❖ Students are advised to choose innovative and challenging definitions.
- ❖ Batch wise project definitions must be unique.
- ❖ Any kind of management system would not be encouraged.
- ❖ Tools like GitHub would be used to track the progress of project development by the concern faculty. Concerned guide will demonstrate the working of GitHub Tool.
- ❖ Student has to prepare Report/Paper at end of semester as part of submission.
- ❖ Report/Paper structure is finalized for semester end submission.
- ❖ To have a better outcome as well as progress tracking at the end of semester, it isdecided that students have to appear for internal reviews, which will help them to get more insight in the project.
- ❖ To maintain similarity below 40%, Students have to submit project's final document to concern SGP guide for plagiarism check before 15 days of external exam.
- ❖ Students have to attach plagiarism report in final spiral bound with duly signed by SGP guide.
- ❖ Students have to bring internal review card hard copy on the day of internal review exam, after that they will bring filled review card on the day of external review.

D. Student Learning Outcome:

After learning the course, students will able to

CO1	Identify a range of solutions, critically evaluate and justify proposed design solution.
CO2	Manage learning & self-development including development of organizational skills, time management, effective use of scientific literature and discriminating use of Web resources.
CO3	Apply a wide range of principles and tools available to the software developer such as choice of the algorithm, language, software libraries etc.
CO4	Write and test programs using appropriate test cases.
CO5	Solve communication issues in large, complex software projects andStructure & communicate ideas effectively orally. Also Prepare & deliver coherent and structured verbal and written technical reports.
CO6	Evaluate system in terms of general quality attributes and possible trade-offs presented within the given problem/system.

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12	PSO1	PSO2

CO1	3	3	1	2	-	2	2	2	2	-	1	3	3	2
CO2	3	3	1	2	1	2	2	2	2	-	1	3	3	2
CO3	3	1	3	3	3	1	2	2	3	-	2	3	3	2
CO4	3	1	1	3	1	-	-	1	2	-	2	2	2	2
CO5	3	-	-	-	-	-	2	3	3	3	3	2	2	1
CO6	3	2	1	2	1	-	-	1	2	-	1	1	3	1

E. Recommended Study Material:

- **Reference book:**
  1. John M Nicolas, Project Management for Business, Engineering and Technology, Elsevier.
  2. Sanjay Mohapatra, Software Project Management, Cengage Learning
  3. Clive L. Dym, Patrick Little, Elizabeth J. Orwin, "Engineering Design – A Project Based Introduction", Wiley India Pvt. Ltd.
  4. Hughes & M. Cotterell, "Software Project Management", Tata McgrawHills.
- **Web Materials:**

  1. https://status.net/templates/project-report/
  2. https://www.tutorialspoint.com/software_engineering/software_project_management.htm
  3. https://www.geeksforgeeks.org/coding-standards-and-guidelines/
  4. https://www.altexsoft.com/blog/engineering/8-ways-to-improve-software-testing-through-planning-work-environment-automated-testing-and-reporting/