Faculty of Technology & Engineering
Bachelor of Technology Programme
Computer Engineering
(B.Tech. CE)

ACADEMIC REGULATIONS & SYLLABUS

(Choice Based Credit System)

Academic Year: 2020-2021

Chandubhai S. Patel Institute of Technology

Vision

To become a leading institute for creation and dissemination of knowledge in the frontiers of Technology.

Mission

To Prepare world-class technocrats and researchers and facilitate enhanced deployment of technology for betterment of lives.

U & P U. Patel Department of Computer Engineering

Vision

To teach (impart) the basic fundamentals of engineering concepts (knowledge), technical/technological skills, research skills and above all the practical ability to apply the knowledge to real situations. Additionally, to install the critical values that will help in preparing the students develop into more effective, productive and focused IT professionals, good citizen and industry ready product to touch their lives tomorrow.

Mission

The U & P U. Patel department of Computer Engineering is committed

1. To provide world-class educational activities
2. To deploy latest technologies
3. To foster research activities in consultation with academia and industry

THE PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO1	To prepare the student(s) for successful career as an engineer, a corporate or a government professional, a scientist, ac academician, a technocrat, an administrator and an entrepreneur.
PEO2	To make students demonstrate their abilities to adapt to a rapidly changing environment by having learning approach and apply new skills and new technologies to solve the problems.
PEO3	To create an ambience where the students are cared for in every aspect and motivated to become excellent working professionals who will continue to cherish their association with the organization as a whole, staff and colleagues.
PEO4	To provide continued professional development and lifelong learning throughout their career to inculcate strong teamwork and by installing lacking skills for the benefit of the society.
PEO5	To prepare the students to apply their technical skill(s) to analyse and design appropriate solution(s) with social consciousness and ethical values.

PROGRAM ARTICULATION MATRIX

(-Not Mathing,1-Low,2,3-High)

Mission Statement	PEO1	PEO2	PEO3	PEO4	PEO5
To provide world-class educational activities	3	3	3	3	3
To deploy latest technologies	-	2	3	3	-
To foster research activities in consultation with academia and industry	3	2	2	3	3

PROGRAM OUTCOMES (POs)

At the end of the program, the student will be able to:

PO1	**Engineering knowledge:** Apply knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet specified needs with appropriate consideration for public health and safety, and the cultural, societal, and environmental considerations.
PO4	**Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	**The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	**Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	**Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	**Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	**Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

At the end of the program, the student will be able to:

PSO1	Apply good analytical, design and implementation skills required to formulate and solving computational problems.
PSO2	Excellent adaptability to function in multi-disciplinary work environment, good interpersonal skills in appreciation of professional ethics and societal responsibilities.

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 – 1 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 = $\sum C_i G_i / \sum C_i$ 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 = $\sum C_i G_i / \sum C_i$ 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.

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											
TEACHING & EXAMINATION SCHEME FOR B TECH PROGRAMME IN CE ENGINEERING											
CHOICE BASED CREDIT SYSTEM											
Sem	Course Code	Course Title	Teaching Scheme			Credit	Examination Scheme				
			Contact Hours				Theory		Practical		Total
			Theory	Practical	Total		Internal	External	Internal	External	
Sem 3	MA253	Discrete Mathematics	4	0	4	4	30	70	0	0	100
	CE251	Java Programming	3	4	7	5	30	70	50	50	200
	CE252	Digital Electronics	3	2	5	4	30	70	25	25	150
	CE257	Data Communication & Networking	4	2	6	5	30	70	25	25	150
	XXXXX	University Elective - I	2			2	30	70			100
	CE244	Software Group Project-I	0	4	4	2			50	50	100
	HS121.02 A	Creativity Problem Solving & Innovation	2	0	2	2					100
			18	**12**	**30**	**24**	**120**	**280**	**180**	**220**	**900**
Sem 4	CE245	Data Structures & Algorithms	3	2	5	4	30	70	25	25	150
	CE246	Database Management System	4	4	8	6	30	70	50	50	200
	CE258	Microprocessor and Computer Organization	4	2	6	5	30	70	25	25	150
	CE259	Programming in Python	0	2	2	1	0	0	25	25	50
	XXXXX	University Elective - II	2			2			30	70	100
	CE255	Software Group Project-II	0	4	4	2			50	50	100
	HS111.02 A	Human Value & Professional Ethics	2	0	2	2					100
			15	**14**	**29**	**22**	**90**	**210**	**205**	**245**	**850**

Note:

- **University Elective (UE):-** University Electives are offered in common slots and offered by various departments. Students of any programme can select these electives. Subjects like Research Methodology, Occupational Health & Safety, Engineering Economics, Professional Ethics, and Project Management, Disaster Management, Risk Management etc. can be included.

- **Cluster Elective (CT):-** Institutional Electives means common electives among a cluster of programmes (eg. CE/IT/EC/EE etc.). If Institutional Electives are not applicable, it will be Programme electives
- **Programme Elective (PE):-**
- **Institute Elective (IE):-**

- Provision for Auditing a course will be available
- Audit courses may be offered and decided based on need of the institute/program(s)

University Elective - I (UE - I)	University Elective - II (UE - II)
Introduction to MATLAB Programming	Prototyping Electronics with Arduino
Art of Programming	Web Designing
Environmental Sustainability and Climate Change	Basics of Environmental Impact Assessment
Python for Electrical Engineering	Computer Programming for Electrical Engineering
ICT Resources and Multimedia	Internet Technology and Web Design
Engineering Drawing	Material Science
Fundamentals of Packaging	Cosmetics in daily life
Basic Laboratory Techniques	Life Style Diseases & Management
First Aid & Life Support	Occupational Health & Ergonomics
Health Promotion and Fitness	Programming the Internet
Introduction to Web Designing	Health Care Management
Banking and Insurance	

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY											
TEACHING & EXAMINATION SCHEME FOR B TECH PROGRAMME IN CE ENGINEERING											
CHOICE BASED CREDIT SYSTEM											
Sem	Course Code	Course Title	Teaching Scheme				Examination Scheme				
			Contact Hours			Credit	Theory		Practical		Total
			Theory	Practical	Total		Internal	External	Internal	External	
Sem 5	CE341	Microprocessor architectures and assembly programming	3	2	5	4	30	70	25	25	150
	CE342	Design & Analysis of Algorithms	3	2	5	4	30	70	25	25	150
	CE343	Software Engineering	3	2	5	4	30	70	25	25	150
	CE344	Computer Networks	3	2	5	4	30	70	25	25	150
	CEXXX	Programme Elective-I	2	4	6	4	30	70	50	50	200
	CE352	Software Group Project-III	0	2	2	2			50	50	100
	CE346	Summer Internship-I			0	3	0	0	75	75	150
	HS131.02 A	Communication and Soft skills	0	2	2	2	0	0	30	70	100
			14	**16**	**30**	**27**	**150**	**350**	**305**	**345**	**1150**
Sem 6	CE347	Internals of Operating |System	3	2	5	4	30	70	25	25	150
	CE348	Information Security	4	2	6	5	30	70	25	25	150
	CE349	Theory of Computation	3	0	3	3	30	70	0	0	100
	CE350	Data Warehousing & Data Mining	3	2	5	4	30	70	25	25	150
	CEXXX	Programme Elective-II	3	2	5	4	30	70	25	25	150
	CE353	Software Group Project-IV	0	4	0	4			100	100	100
	HS132.02 A	Contributory Personality Development	0	2	2	2	30	70	0	0	100
			16	**14**	**30**	**26**	**180**	**420**	**200**	**200**	**1000**

Note:

- **University Elective (UE):-** University Electives are offered in common slots and offered by various departments. Students of any programme can select these electives. Subjects like Research Methodology, Occupational Health & Safety, Engineering Economics, Professional Ethics, and Project Management, Disaster Management, Risk Management etc. can be included.

- **Cluster Elective (CT):-** Institutional Electives means common electives among a cluster of programmes (eg. CE/IT/EC/EE etc.). If Institutional Electives are not applicable, it will be Programme electives
- **Programme Elective (PE):-**

- **Institute Elective (IE):-**
- Provision for Auditing a course will be available
- Audit courses may be offered and decided based on need of the institute/program(s)

List of Programme Electives			
Code	**Programme Elective - I (PE - I)**	**Code**	**Programme Elective - II (PE - II)**
CE371	Advanced Java programming	CE374	Service Oriented Computing
CE372	Advanced Programming using .NET Framework	CE375	Digital Image Processing
CE373	Mobile Application Developments	CE376	Programming in Python

List of HSS Electives	
Code	**HSS Elective - III**
HS125.01 A	Society, Governance and International Studies
HS130 A	Law & Justice
HS134 A	Contributor Personality Development

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY											
TEACHING & EXAMINATION SCHEME FOR B TECH PROGRAMME IN CE ENGINEERING											
CHOICE BASED CREDIT SYSTEM											
Sem	Course Code	Course Title	Teaching Scheme				Examination Scheme				
			Contact Hours			Credit	Theory		Practical		Total
			Theory	Practical	Total		Internal	External	Internal	External	
Sem 7	CE441	Big Data Analytics	3	4	7	5	30	70	50	50	200
	CE442	Design of Language Processors	4	2	6	5	30	70	25	25	150
	CE443	Cloud Computing	3	2	5	4	30	70	25	25	150
	CE444	Internet of Things	3	2	5	4	30	70	25	25	150
	CEXXX	Programme Elective-III	4	2	6	5	30	70	25	25	150
	CE448	Software Group Project-V	0	2	2	1	0	0	25	25	50
	CE446	Summer Internship-II			0	3	0	0	75	75	150
			17	**14**	**31**	**27**	**150**	**350**	**250**	**250**	**1000**
Sem 8	CE447	Software Project Major	0	36	36	20			250	350	600

Note:

- **University Elective (UE):-** University Electives are offered in common slots and offered by various departments. Students of any programme can select these electives. Subjects like Research Methodology, Occupational Health & Safety, Engineering Economics, Professional Ethics, and Project Management, Disaster Management, Risk Management etc. can be included.

- **Cluster Elective (CT):-** Institutional Electives means common electives among a cluster of programmes (eg. CE/IT/EC/EE etc.). If Institutional Electives are not applicable, it will be Programme electives
- **Programme Elective (PE):-**
- **Institute Elective (IE):-**
- Provision for Auditing a course will be available
- Audit courses may be offered and decided based on need of the institute/program(s)

LIST OF ELECTIVE SUBJECTS FOR FOURTH YEAR B TECH PROGRAMME IN COMPUTER ENGINEERING		
ELECTIVES	Code	Elective - III
	CE471	Block chain Technology
	CE472	Wireless Communication & Mobile Computing
	CE473	Machine Learning

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 3)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

MA253: DISCRETE MATHEMATICS AND ALGEBRA

Credits and Hours:

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

A. Objective of the Course:

Discrete Mathematics and Algebra have many applications in Computer Engineering and Information Technology. This course contains many concepts which are applicable to subjects like Theory of Computation, Artificial Intelligence, Data Structure and Algorithms, Compiler Constructions, Algorithm Analysis and Design, Digital Electronics etc.

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 (Theory): 60

C. Detailed Syllabus:

1.	**Predicate Calculus:**	**08 Hours**	**13%**
	Revision: Propositions, connectives, converse, inverse, contrapositive, tautology, contradiction. Logical equivalence. Minimal functionally complete set of connectives. Principle conjunctive normal forms and Principle disjunctive normal forms. Predicate calculus using rules of inferences.		
2.	**Relations and Lattice:**	**10 Hours**	**17%**

	Revision of properties of relations on sets. Representations of relations: graphical and matrix representation. Equivalence relation, covering of a set, partition of a set. Partially ordered sets, totally ordered sets, Hasse diagram. Lattices, sub lattices. Properties of lattices (without proof). Complete lattices, bounded lattices, distributive lattices, complemented lattices and complemented distributive lattices.		
3.	**Graph Theory:**	**12 Hours**	**20%**
	Basic terminologies, Simple graph, Types of graphs. Degree of a vertex, matrix representations of graph. Path and connectivity. Eulerian and Hamiltonian graph. Subgraphs, spanning subgraphs, isomorphic graphs. Planar graphs. Matching in graphs. Graph coloring.		
4.	**Recurrence Relations:**	**05 Hours**	**08%**
	Solutions of recurrence relation by direct methods. Generating functions and solutions of recurrence relation.		
5.	**Abstract Algebra:**	**09 Hours**	**15%**
	Groupoid, semi group, monoid, group. Order of group, order of an element, Lagrange's theorem. Subgroup, cyclic subgroup, permutation group.		
6.	**Linear Algebra:**	**16 Hours**	**27%**
	Vector space: definition and examples. Subspaces. Linear combinations, linearly dependence and linearly independence. Basis and dimension of a vector space. 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 subject must be discussed.
- Lectures may be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lectures/laboratory which carries a 5% component of the overall evaluation.
- Minimum two internal tests/ unit tests must be conducted and average of two will be considered as a part of 15% overall 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%.
- 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 and lattices on sets.
CO3	Graph and types of the graphs and identify the real world phenomena in terms of graph theory.
CO4	The concept of recurrence, generating functions and their applications in solving recurrence relations.
CO5	Different algebraic structures like groupoid, semi group, monoid, group , cyclic group and permutation 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 linear transformation.

- 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. H. Anton and C. Rorres; Elementary Linear Algebra, Application version, Wiley Edition 2010.
2. Kenneth H. Rosen and Kamala Krithivasan. Discrete mathematics and its applications. Vol. 6. New York: McGraw-Hill, 1995.
3. Tremblay, Jean-Paul, and Rampurkar Manohar. Discrete mathematical structures with applications to computer science. New York: McGraw-Hill, 1975

❖ **Reference Books:**

1. Thomas H. Cormen, E. E. Leiserson, R. L. Rivest and C. Stein, Introduction to algorithms (Vol. 6). Cambridge: MIT press,2001
2. Narsingh Deo, Graph theory with applications to engineering and computer science. Courier Dover Publications, 2016.
3. B. Kolman and R. C. Busby, Discrete Mathematical Structures for Computer Science, 2nd edition, Prentice-Hall, Englewood Cliffs, New Jersey 1987.
4. Swapan Kumar Sarkar, A Text Book of Discrete Mathematics, S. Chand and Co. New Delhi 2008.
4. D. S. Malik and Mridul K. Sen. Discrete mathematical structures: theory and applications. Course Technology, 2004.
5. D. F. McAllister and D. F. Stanat. Discrete Mathematics in Computer Science. Prentice-Hall, Inc. 1977.

❖ **Web 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://www-sop.inria.fr/members/Frederic.Havet/Cours/coloration.pdf

Video Lectures:

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

CE251: JAVA PROGRAMMING

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	4	0	7	**5**
Marks	100	100	0	200	

Pre-requisite courses:

- Basic programming skill.

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Fundamental of Programming in Java	02
2.	Class Fundamentals	04
3.	Array & String Handling	02
4.	Inheritance, Interfaces & Packages	06
5.	Exceptions Handling	05
6.	Multithreaded Programming	07
7.	File I/O and NIO	07
8.	Java Collection Frameworks and Generics	12
	Total hours (Theory) :	45
	Total hours (Lab) :	60
	Total hours :	120

Detailed Syllabus:

1.	**Fundamental of Programming in Java**	**02 Hours**	**04%**
	History of Java, Basic overview of java, Bytecode, JVM Buzz-words, Application and applets, Constants, Variables & Data Types, Comment, 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**	**02 Hours**	**04%**
	Array basics, String Array, String class, StringBuffer and StringBuilder class, String Tokenizer Class and Object Class		
4.	**Inheritance, Interfaces & Packages**	**06 Hours**	**14%**
	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, Default Methods, Implementing Interface, Variables in Interface		
5.	**Exceptions Handling**	**05 Hours**	**11%**
	Exception types, Try …Catch…Finally, Throw, Throws, creating your own exception subclasses		
6.	**Multithreaded Programming**	**07 Hours**	**16%**
	Life cycle of thread, thread methods, thread priority, thread exceptions, Implementing Runnable interface, Synchronization and Concurrency		
7.	**File NIO**	**07 Hours**	**16%**
	File and Directories, Byte streams and character streams, Random Access Files,NIO: Meta Data File Attributes Buffers, Channels, Recursive Operation.		
8.	**Collection Framework and Generics**	**12 Hours**	**26%**
	Collections of objects, Collections: Sets, Sequence, Map, Understanding Hashing, Use of Array List & Vector, Generics Class, Lamda Expression, Functional Reference, Method Reference, Optional Classes, Processing data with streams		

Course Outcome (COs):

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

CO1	Understand and implement Object Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity and garbage collection for saving resources.
CO2	Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object-oriented computer language (classes, objects, methods with parameters, abstract classes, interfaces, inheritance and polymorphism).
CO3	Use and develop Concurrency theory: progress guarantees, deadlock, livelock, starvation, linearizability.
CO4	Build and test program using new IO api and exception handling
CO5	Analyze and apply collection framework and generics to solve different data structure algorithms.
CO6	Understand and apply java new features

Course Articulation Matrix:

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

Recommended Study Material:

- **Text Books:**
  1. Java: The Complete Reference, Ninth Edition by Herbert Schildt , Oracle Press.
  2. Java 8 in Action: Lambdas, Streams, and Functional-style Programming by Alan Mycroft and Mario Fusco, Manning Publication

❖ **Reference Books:**

1. Thinking in Java, Bruce Eckel, Prentice Hall
2. Java: A Beginner's Guide (Sixth Edition) by Herbert Schildt , Oracle Press.
3. Core Java Volume I--Fundamentals (9th Edition) (Core Series) by Cay S. Horstmann, Prentice Hal

❖ **Web Materials:**

1. https://docs.oracle.com

CE252: DIGITAL ELECTRONICS

Credits and Hours:

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

Pre-requisite courses:

- Basic Electronics

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Number Systems	5
2.	Boolean Algebra and Logic Gates	5
3.	Simplification of Boolean Functions	6
4.	Combinational Logic	5
5.	Combinational Logic With MSI AND LSI	5
6.	Sequential Logic	10
7.	Registers, Counters and the Memory Unit	5
8.	Processor Logic Design	4
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Number Systems**	**05 Hours**	**10%**
	Digital computer and digital systems, Binary Number, Number base conversion Octal and Hexadecimal Number, Complements, Binary Codes, Binary Storage and register, Binary Logic, Integrated Circuit		
2.	**Boolean Algebra and Logic Gates**	**05 Hours**	**10%**
	Basic Definition, Axiomatic Definition of Boolean Algebra,		

	Minterm And Maxterms, Basic Theorem and Properties of Boolean Algebra, Logic Operations, Digital Logic Gates, IC digital Logic Families		
3.	**Simplification of Boolean Functions**	**06 Hours**	**12%**
	Two-Three Variable K-map, Four- Five Variable K-map, Product of sum Simplification, NAND or NOR implementation, Don't Care condition 3.4 Tabulation method		
4.	**Combinational Logic**	**05 Hours**	**12%**
	Introduction, Design Procedure, Hazards, Adder, subtractor, Code Conversion, Universal Gate, exclusive OR & equivalence functions		
5.	**Combinational Logic With MSI and LSI**	**05 Hours**	**12%**
	Introduction, Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoder, Multiplexer, ROM, PLA, PAL		
6.	**Sequential Logic**	**10 Hours**	**22%**
	Introduction, RS,JK,D,T Flip-Flops, Triggering of Flip-Flops, Flip-Flop Excitation Tables, Analysis of Clocked Sequential Circuits, State Reduction and Assignment Design Procedure, Design of Counters, Design with State Equations		
7.	**Registers, Counters and the Memory unit**	**05 Hours**	**12%**
	Introduction, Registers, Shift Registers, Ripple Counters, Synchronous Counters, Timing Sequences, Memory Unit, Johnson counter		
8.	**Processor Logic Design**	**04 Hours**	**10%**
	Processor Organization, Arithmetic Logic Unit, Design of ALU, Status Register, Design of Shifter, Processor Unit		

Course Outcome (COs):

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

CO1	Understand, convert and examine the structure of various number systems and its application in digital design.
CO2	Simplification of Boolean function and its conversion in different forms.

CO3	Understand, analyse and design various combinational circuits.
CO4	Compare different flip-flop characteristics and design of Flip-Flop with gates. With FF Design, understand and analyse various sequential circuits.
CO5	Explain the basic requirements of ALU design and its variation.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Digital Logic and Computer Design By M Morris Mano, PHI- Publication 2002.
  2. Digital Principles and Applications By Malvino & Leach, Seventh Edition, McGraw-Hill Education.
- **Reference book:**
  1. Digital Design M. Morris Mano and Michael D. Ciletti, Pearson Education.

CE257: DATA COMMUNICATION & NETWORKING

Credits and Hours:

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

Pre-requisite courses:

- N/A

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction and Basic Concepts	04
2.	Network Model	04
3.	Signals	04
4.	Digital and Analog Transmission	09
5.	Multiplexing	06
6.	Data Transmission	02
7.	Error Correction and Detection	05
8.	Data Link Control	04
9.	Media Access Control	06
10.	Switching	01
11.	Wired and Wireless LAN	09
12.	Network Routing	06
	Total hours (Theory) :	60
	Total hours (Lab) :	24
	Total hours :	84

Detailed Syllabus:

1.	**Introduction and Basic Concepts**	**04 Hours**	**6%**
	Data Communication, Networks, Network Types, Internet History, Standards and Administration		
2.	**Network Model**	**04Hours**	**6%**

	Protocol Layering, TCP/IP Protocol Suites, The OSI Model		
3.	**Signals**	**04 Hours**	**6%**
	Data and Signals, Periodic Analog Signal, Digital Signal, Transmission Impairment, Data Rate limits, Performance		
4.	**Digital and Analog Transmission**	**09 Hours**	**15%**
	Transmission Modes, Digital to Digital Conversion, Analog to Digital Conversion, Digital to Analog Conversion, Analog to Analog Conversion		
5.	**Multiplexing**	**06 Hours**	**10%**
	Frequency division Multiplexing, Wave length division Multiplexing, Time division Multiplexing, Multiplexing applications, Spread Spectrum		
6.	**Data Transmission**	**02 Hours**	**3%**
	Guided Media, Unguided Media,		
7.	**Error Correction and Detection**	**05 Hours**	**8%**
	Types of Errors, Redundancy, Detection versus correction, Block Coding, Cycle Coding, Checksum		
8.	**Data Link Control**	**04 Hours**	**6%**
	DLC Services, Data Link Layer Protocol, HDLC, Point-to-point Protocol		
9.	**Media Access Control**	**06 Hours**	**10%**
	Random Access, Controlled Access, Channelization		
10.	**Switching**	**01 Hours**	**1%**
	Circuit Switching, Packet Switching		
11.	**Wired and Wireless LAN**	**09Hours**	**15%**
	Ethernet Protocol, Standard Ethernet, Fast Ethernet, Gigabit Ethernet, Wireless characteristics and access control, IEEE 802.11 project.		
12	**Network Routing**	**06 Hours**	**10%**
	Unicast Routing, Routing Algorithms, Unicast Routing Protocols: IS,RIP,OSPF		

Course Outcome (COs):

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

CO1	Understand and identify different physical layer transmission fundamentals such as types of signals, transmission, multiplexing, types of medium and modulation.
CO2	Evaluate existing layer-2 networking standards and implementations.
CO3	Evaluate key networking protocols, and their hierarchical relationship in the context of a conceptual model, such as the OSI and TCP/IP framework.
CO4	Understand existing different medium access protocols and evaluate for adoption for future networking.
CO5	Understand and differentiate functionality of existing network routing protocols.
CO6	Measure different network parameter such as Throughput & different types of delays.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Data communication & Networking, BahrouzForouzan, McGraw-Hill
- **Reference book:**
  1. Data and Computer Communications, William Stallings, Prentice Hall

2. Computer Network, Andrew S. Tanenbaum, Fourth Edition, Prentice Hall

❖ **Web material:**

1. www.wikipedia.org
2. http://www.webopedia.com

❖ **Software:**

1. Wireshark
2. Cisco Packet Tracer

CE244: SOFTWARE GROUP PROJECT-I

Credits and Hours:

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

Pre-requisite courses:

- Programming Language, Software Engineering.

Outline of the Course:

- Student at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students will work together in a team (at most three) with any programming language.
- 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 of continuous 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. Final Project Report
  b. Project Setup file with Source code
  c. Project Presentation (PPT)
- A student has to produce some useful outcome by conducting experiments or project work.

Total hours (Theory) : 00

Total hours (Lab) : 30

Total hours : 30

Course Outcome (COs):

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

CO1	Create enhanced employment by moulding the students with higher technical skill.
CO2	Promote creative thinking, to provide hands on actual technology.
CO3	Handle software project and get use to software development processes.
CO4	Correlate knowledge of different subjects and apply theoretical knowledge to implement project for identified problem.
CO5	Write technical report and deliver presentation by applying different visualization tools and evaluation metrics.
CO6	Improve communication and presentation skill.

Course Articulation Matrix:

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

Recommended Study Material:

- **Reference book:**
  1. Books, Magazines ,Journals & online course platforms of related topics
- **Web material:**
  1. www.ieeexplore.ieee.org
  2. www.sciencedirect.com
  3. www.elsevier.com

4. https://www.udemy.com/
5. https://www.udacity.com/
6. https://nptel.ac.in/course.html
7. https://www.futurelearn.com/

❖ **Software:**

1. ASP.NET
2. PYTHON/MATLAB
3. PHP
4. ANDROID/IOS

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 4)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE245: DATA STRUCTURE AND ALGORITHMS

Credits and Hours:

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

Pre-requisite courses:

- Programming Language

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.	Sorting	10
5.	Searching	01
6.	Dictionaries	02
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Introduction**	**04 Hours**	**08%**
	Introduction to data structure (Types of data structure), Introduction to algorithms. Algorithm Analysis and Big O notation, Memory representation of Array: Row Order and Column Order, Abstract Data Types(ADT)		
2.	**Linear Data Structure**	**12 Hours**	**27%**
	Stack: Operations: push, pop, peep, change, Applications of Stack: Recursion: Recursive Function Tracing, Principles of recursion, Tail recursion, Removal of Recursion, Tower of		

	Hanoi, Conversion: Infix to Postfix, Infix to Prefix. Evaluation: Prefix and Postfix expression, **Queue** Simple Queue: Insert and Delete operation, Circular Queue: Insert and Delete operation, Concepts of: Priority Queue, Double-ended Queue, Applications of Queue, **Linked List:** Memory Representation of LL, Singly Linked List, Doubly Linked List ,Circular Linked List ,Applications of Linked List		
3.	**Non Linear Data Structure**	**16 Hours**	**36%**
	Tree: Tree Concepts, Tree Traversal Techniques: Pre-order, Post-order and In-order (Recursive and Iterative), Binary Search Tree: Iterative and Recursive, Balanced Trees (AVL Trees, Applications of Tree, **Heaps**: priority queues and Binary Heaps, **Graph:** Graph concepts, Memory Representation of Graph, BFS and DFS, Applications of Graph		
4.	**Sorting**	**10 Hours**	**23%**
	Sorting (concepts, Selection Sort, Bubble Sort, Merge Sort, Radix Sort, Insertion Sort, Heap Sort, Quick Sort)		
5.	**Searching**	**01 Hours**	**02%**
	Sequential Search, Binary Search		
6.	**Dictionaries**	**02 Hours**	**04%**
	Hashing, Hashing Functions, Collision-Resolution Techniques, Applications		

Course Outcome (COs):

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

CO1	Understand and Implement Algorithms and core Data Structures such as stack, queue, hash table, priority queue, binary search tree and graph in programming language.
CO2	Analyse data structures in storage, retrieval and computation of ordered or unordered data.
CO3	Compare alternative implementations of data structures with respect to demand

	and performance.
CO4	Describe and evaluate the properties, operations, applications, strengths and weaknesses of different data structures.
CO5	Apply and select the most suitable data structures to solve programming challenges.
CO6	Discover advantages and disadvantages of specific algorithms.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. An Introduction to Data Structures with Applications, Jean-Paul Tremblay, Paul G. Sorenson, McGraw-Hill.
  2. Data structure with C, Lipschutz, TMH
  3. Introduction to Algorithms: Cormen, Leiserson, Rivest and Stein: Prentice Hall of India
  4. Data Structures and Algorithms: Aho, Hopcroft and Ullmann: Addison Wesley.

- **Reference book:**
  1. Classic Data structures, D.Samanta, Prentice-Hall International.0

2. Data Structures using C & C++, Ten Baum, Prentice-Hall International.
3. Data Structures: A Pseudo-code approach with C, Gilberg & Forouzan, Thomson Learning.
4. Fundamentals of Data Structures in C++, Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, W. H. Freeman.
5. "A Practical Introduction to Data Structures and Algorithm Analysis" by Clifford A. Shaffer
6. Data Structures and Algorithm in Java: Goodrich and Tamassia: John Wiley and Sons.

❖ **Web material:**

1. http://www.leda-tutorial.org/en/official/ch02s02s03.html
2. http://www.leda-tutorial.org/en/official/ch02s02s03.html
3. http://www.softpanorama.org/Algorithms/sorting.shtml

CE246: Database Management System

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	4	4	-	8	**6**
Marks	100	100	-	200	

Pre-requisite courses:

- Data Structure

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introductory concepts of DBMS	04
2.	Relational Model	06
3.	Entity-Relationship model	07
4.	Formal Relational Query Languages	06
5.	Relational Database Design	09
6.	Transaction & Recovery Management	07
7.	Advanced Transaction Processing	06
8.	Database Security	06
9.	Indexing and Hashing	07
10.	Query Processing & Query Optimization	02
	Total hours (Theory) :	60
	Total hours (Lab) :	60
	Total hours :	120

Detailed Syllabus:

1.	**Introductory concepts of DBMS**	**04 Hours**	**06%**
	Introduction and applications of DBMS, Purpose of database, Data Independence, Database System architecture- levels, Mappings, Database users and DBA		
2.	**Relational Model**	**06 Hours**	**10%**
	Structure of Relational Databases, Database Schema,		

	Schema Diagram, Domains , Relations, Relational Query Languages, Relational Operations		
3.	**Entity-Relationship model**	**07 Hours**	**12%**
	Basic concepts, Design process, Constraints, Keys, Design issues, E-R diagrams, Weak Entity Sets, Extended E-R features- Generalization, Specialization, Aggregation, Reduction to E-R database schema		
4.	**Formal Relational Query Languages**	**06 Hours**	**10%**
	The relational Algebra, The Tuple Relational Calculus, The Domain Relational Calculus		
5.	**Relational Database design**	**09 Hours**	**15%**
	Functional Dependency–definition, Trivial and Non-Trivial FD, Closure of FD set, Closure of attributes, Irreducible set of FD, Normalization – 1NF, 2NF,3NF, Decomposition using FD- Dependency Preservation		
6.	**Transaction & Recovery Management**	**07 Hours**	**12%**
	Transaction concepts, Properties of Transactions, Serializability of transactions, Testing for Serializability, System recovery, Two- Phase Commit protocol, Recovery and Atomicity, Log-based recovery¸ Concurrent executions of transactions and related problems, Locking mechanism, Solution to Concurrency Related Problems, Deadlock, Two-phase locking protocol, Intent locking		
7.	**Advanced Transaction Processing**	**06 Hours**	**10%**
	Transaction-Processing Monitors, Transactional Workflows, Main-Memory Databases, Real-Time Transaction Systems, Long-Duration Transactions		
8.	**Database Security**	**06 Hours**	**10%**
	Views - What are views for?, View retrievals, View updates, Snapshots (a digression), Materialized view, Security – Security and Authentication, authorization in SQL, Data encryption, Missing Information - An overview of the 3VL approach		

9.	**Indexing and Hashing**	**07 Hours**	**12%**
	Basic Concepts, Ordered Indices, B+-Tree Index Files, B+-Tree Extensions, Multiple-Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL		
10.	**Query Processing & Query Optimization**	**02 Hours**	**03%**
	Overview, Measures of Query Cost, Selection Operation, Sorting, Join, Evaluation of Expressions, Transformation of relational Expressions, Estimating Statistics of expression results, Query Evaluation plans		

Course Outcomes (COs):

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

CO1	Apply the concepts of engineering i.e collecting data, organize the data in the systematic form, arrange the data in a computational way and applying mathematics formation.
CO2	Analyse how data are stored and maintained using data models. Ready to assimilate the concept of data abstraction and design queries using SQL. Identify how data is represented in the relational model and create relations using SQL language
CO3	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.
CO4	Produce aggregate operators to write SQL queries which are not expressible in relational algebra. “More mathematical” notation may apply and also used in research and other venues. Combining these concepts allows production of sophisticated queries.
CO5	Decompose un-normalized tables into normalized compliant tables. Design and implement a normalize database schema for a given problem-domain.
CO6	Compare transactions and their properties with (ACID) and without ACID. Apply locking protocol to ensure isolation. Develop logging technique to ensure atomicity and durability

CO7	Design a logical view which can be used for analytical tasks. Develop practical experience of the design and implement scalable, secure databases.
CO8	Produce strategies to minimise risks of security breaches in a range of network environments and data storage systems. Compute retrieval time and concluding with suitable indexing technique
CO9	Compare and evaluate query execution plan.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	1	2	1	1	-	-	-	1	2	-
CO2	3	3	3	2	3	1	2	-	-	-	-	3	2	1
CO3	3	3	3	3	2	2	1	3	2	1	-	2	3	2
CO4	3	3	1	3	1	1	-	2	2	-	-	2	2	2
CO5	2	2	3	2	3	1	2	2	1	-	-	3	1	1
CO6	3	3	2	1	-	2	-	-	1	-	-	2	-	1
CO7	3	3	3	2	3	1	-	2	2	1	-	3	2	2
CO8	2	3	2	2	2	2	2	3	1	-	-	3	3	1
CO9	2	2	3	1	3	-	-	1	1	-	-	3	3	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 "-"

Recommended Study Material:

- **Text book:**
  1. Database System Concepts, Abraham Silberschatz, Henry F. Korth& S. Sudarshan, McGraw Hill.
  2. An introduction to Database Systems, C J Date, Addition-Wesley
- **Reference book:**
  1. "Fundamentals of Database Systems", R. Elmasri and S.B. Navathe, the Benjamin / Cumming Pub. Co

2. SQL,PL/SQL the Programming Language of oracle, Ivan Bayross, BPB Publications
3. Oracle: The Complete Reference, George Koch, Kevin Loney, TMH /oracle press

❖ **Web material:**

1. http://www.sql.org
2. http://www.w3schools.com
3. http://www.sqlcourse.com

❖ **Software:**

1. Oracle 10g
2. SQL Lite
3. Live SQL
4. Firebase
5. Squirrel SQL
6. Postgre SQL

CE258: Microprocessor and Computer Organization

Credits and Hours:

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

Pre-requisite courses:

Digital Electronics

A. Outline of the course:

Sr. No.	Title of the unit	Minimum number of hours
1	Introduction to digital logic Circuit	03
2	Register Transfer and Microoperations	09
3	Basic Computer Organization and Design	08
4	**Central Processing Unit**	05
5	Pipeline and Vector Processing	05
6	Computer Arithmetic	06
7	Memory Organization	06
8	8086,80186, 80286 Processor	06
9	80386 Processors	10
10	Current Era of Microprocessors	02
	Total hours (Theory).	60
	Practical Hours:	20
	Total hours:	80

.

.

B. Detailed Syllabus:

1.	**Introduction to digital logic Circuit**	**03 Hours**	**07%**
	Digital Computers, Logic Gates, Combinational Circuits (Half adder, Full Adder), Flip-Flops(SR, D, JK, T, Edge-Triggered)		
2.	**Register Transfer and Microoperations**	**09 Hours**	**18%**
	Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Microoperation, Logic Microoperations, Shift Microoperation, Arithmetic Logic Shift Unit.		
3.	**Basic Computer Organization and Design**	**08 Hours**	**18%**
	Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory Reference , Instructions, Input-Output and Interrupt, Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic.		
4.	**Central Processing Unit**	**05 Hours**	**17%**
	Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes.		
5.	**Pipeline and Vector Processing**	**05 Hours**	**09%**
	Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.		
6.	**Computer Arithmetic**	**06 Hours**	**18%**
	Introduction: Binary, Octal, Decimal, Hexadecimal representation, Integer Numbers: Sign-Magnitude,1's complement,2's complement, Addition and Subtraction, Multiplication Algorithm.		
7.	**Memory Organization**	**06 Hours**	**13%**
	Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.		
8.	**8086, 80186, 80286 Processor**	**06 Hours**	**13%**
	Architectural differences of 8086, 80186 and 80286 Processors.		
9.	**80386 Processors**	**10 Hours**	**29%**

	System Architecture, Registers, Memory management: Segment Translation, Page Translation, Combining Segment and Page Translation.		
10.	**Current Era of Microprocessors**	**02 Hours**	**4%**
	Comparison of AMD and Intel Architecture, Features of current era of AMD and Intel processors; Tick-Tock: manufacturing pattern of Intel.		

Course Outcome (COs):

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

CO1	Recognize elements of digital logic circuit. Moving from design of single bit function to multibit function. (Flip flop, Logic Gates, Combinational Circuit). Design circuit for fixed function arithmetic function. Understand the notation of writing register transfer language.
CO2	Design and examine the different Arithmetic, Logic and Shift circuit & Design control unit of Arithmetic, Logic and Shift Circuit.
CO3	Conceptualize and evaluate various parallelism employed in microprocessor.
CO4	Demonstrate and evaluate computer arithmetic operations on integer and real numbers using hardwired algorithm.
CO5	Understand and differentiate n-way set associative memory.
CO6	Understand segment and page translation currently employed in microprocessor. Understand basics of architecture of current era of microprocessors.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1	1	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	3	3	-	-	-	-	1	1	-	-	1	2	-
CO3	3	3	3	1	-	-	-	1	1	-	-	-	2	-
CO4	2	2	3	-	-	-	-	1	1	-	-	-	2	-
CO5	3	3	-	-	-	-	-	-	-	-	-	-	2	-
CO6	3	3	3	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 "-"

Recommended Study Material:

- **Text book:**
  1. Computer System Architecture, Morris Mano (3rd Edition) Prentice Hall.
  2. 80386 Programmer's Reference Manual from MIT.
  3. Microprocessors and Interfacing: Experiments Manual: Programming and Hardware by Douglas V. Hall.
- **Reference book:**
  1. William Stalling, Computer Organization & Architecture-Designing for Performance, Pearson Prentice Hall (8th Edition).
  2. A.S. Tananbum, Structured Computer Organization, Pearson Publisher.
  3. The Essentials of Computer Organization and Architecture Linda Null, Julia Lobur.
  4. John P Hayes, Computer Architecture & Organization, McGraw-Hill.
  5. Computer Architecture: Pipelined and Parallel Processor Design Michael J. Flynn (4th edition).
- **Web Materials:**
  1. www.nptel.iitm.ac.in
  2. https://css.csail.mit.edu/6.858/2014/readings/i386.pdf (80386 Programmer Reference Material)
- **Simulators:**
  1. 8085 &8086 Simulator.

CE259: PROGRAMMING IN PYTHON

Credits and Hours:

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

Pre-requisite courses:

- High level language (C/C++/Java)
- Web Programming

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1	Basics of Python	02
2	Data Structures: Lists, Tuples, Dictionaries and Strings	04
3	Control structures and Function	04
4	Modules and Scoping Rules	02
5	Exceptions Handling	04
6	Magic Methods, Properties, and Iterators	04
7	Object Oriented Programming	06
8	Regular Expression and File Handling	04
	Total hours (Theory) :	0
	Total hours (Lab) :	30
	Total hours :	30

Detailed Syllabus:

1.	**Basics of Python**	**02 Hours**	**07 %**
	Using the Python Interpreter, Variables, Identifiers and Keywords, Numbers and Expressions		
2.	**Data Structures: List, Tuples, Dictionaries and Strings**	**04 Hours**	**13 %**
	Common Sequence Operations: Indexing, Slicing, Adding Sequences, Multiplication, Membership, Length, Minimum, and Maximum, Using Lists as Stacks, Using Lists as Queues, List Comprehensions, Nested List Comprehensions, the del statement,		

	Tuples and Sequences, Sets, Dictionaries, Comparing Sequences and Other Types, Basic String Operations		
3.	**Control Structures and Functions**	**04 Hours**	**13%**
	Conditional Branching: if Statements, break and continue Statements, and else Clauses on Loops, pass Statements Loops: while Loops, for Loops, Defining Functions, More on Defining Functions: Default Argument Values, Keyword Arguments, Arbitrary Argument Lists, Unpacking Argument Lists, Lambda Expressions, Documentation Strings, Function Annotations		
4.	**Modules and Scoping Rules**	**02 Hours**	**07%**
	Executing modules as scripts, The Module Search Path, "Compiled" Python files, Packages: Importing * From a Package, Intra-package References, Packages in Multiple Directories		
5.	**Exception Handling**	**04 Hours**	**13%**
	Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Predefined Clean-up Actions		
6.	**Magic Methods, Properties and Iterators**	**04 Hours**	**13%**
	Constructors, Item Access: The Basic Sequence and Mapping Protocol, Properties: The property Function, Static Methods and Class Methods, __getattr__, __setattr__, and Friends, Iterators, Generators, Generator Expressions		
7.	**Object Oriented Programming**	**06 Hours**	**20%**
	Python Scopes and Namespaces, Class Definition, Class Objects, Instance Objects, Method Objects, Class and Instance Variables, Inheritance, Multiple Inheritance, Private Variables, Polymorphism, Using Properties to Control Attribute Access, Creating Complete Fully Integrated Data Types		
8.	**Regular Expression and File Handling**	**04 Hours**	**14%**
	What is a regular expression?, Regular expressions with special characters, Regular expressions and raw strings, Extracting matched text from strings, Substituting text with regular expressions, Writing and Reading Binary Data, Writing and Parsing Text Files, Iterating		

	over File Contents, Writing and Parsing XML Files, Random Access Binary Files		

Course Outcome (COs):

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

CO1	Interpret the fundamental python syntax, semantics and fluent in the use of python control flow statements. Express proficiency in the handling of strings and functions.
CO2	Determine the methods to create and manipulate python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
CO3	Identify the commonly used operations involving file systems and regular expressions.
CO4	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python along with magic methods.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Magnus Lie Hetland, "Beginning Python From Novice to Professional", Third Edition, Apress,2017
  2. Nigel George, "Mastering Django: Core" Packt Publishing, 2016

❖ **Reference book:**

1. David Beazley, Brian K. Jones, "Python Cookbook", 3rd edition, OREILLY,2016
2. Brett Slatkin, "Effective Python: 59 Specific Ways to Write Better Python", Novatec, 2016
3. Allen Downey, "Think Python: How to Think Like a Computer Scientist", Green Tea Press,2015
4. Mark Lutz "Learning Python", 4th Edition, O'REILLY, 2016
5. Arun Ravindran, Aidas Bendoraitis, Samuel Dauzon, "Django: Web Development with Python",Packt Publishing, 2016

❖ **Web material:**

1. https://www.python.org/
2. http://www.diveintopython3.net/
3. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django
4. https://www.fullstackpython.com/django.html

❖ **Software:**

1. Python IDLE
2. Anaconda Python
3. PyCharm

CE255: SOFTWARE GROUP PROJECT-II

Credits and Hours:

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

Pre-requisite courses:

- Programming Language, Software Engineering.

Outline of the Course:

- Student at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students will work together in a team (at most three) with any programming language.
- 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 of continuous 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.
  d. Final Project Report
  e. Project Setup file with Source code
  f. Project Presentation (PPT)
- A student has to produce some useful outcome by conducting experiments or project work.

Total hours (Theory) :	00
Total hours (Lab) :	30
Total hours :	30

Course Outcome (COs):

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

CO1	Identify problems present in society by surveying variety of domains and convert in project definition.
CO2	Explore new ideas and techniques to solve it. Create, select and apply appropriate techniques, resources, modern engineering and IT tools to solve problem.
CO3	Correlate knowledge of different subjects and apply theoretical knowledge to implement project for identified problem.
CO4	Apply ethical principles and commit to responsibilities and norms of the project.
CO5	Write technical report and deliver presentation by applying different visualization tools and evaluation metrics.
CO6	Apply engineering and management principles to achieve project goal.

Course Articulation Matrix:

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

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines ,Journals & online course platforms of related topics
- **Web material:**
    1. www.ieeexplore.ieee.org
    2. www.sciencedirect.com
    3. www.elsevier.com

4. https://www.udemy.com/
5. https://www.udacity.com/
6. https://nptel.ac.in/course.html
7. https://www.futurelearn.com/

❖ **Software:**

1. ASP.NET
2. PYTHON/MATLAB
3. PHP
4. ANDROID/IOS

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 5)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE341: MICROPROCESSOR ARCHITECTURES AND ASSEMBLY PROGRAMMING

Credits and Hours:

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

Pre-requisite courses:

- Digital Electronics
- Data Communication

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	The Processor: 8086	06
2.	8086 Instruction set and Assembler Directives	06
3.	The Art of Assembly Language Programming with 8086	06
4.	Special Architectural features and related Programming	06
5.	DMA Controllers, Multi-microprocessor systems	03
6.	80286 Processor	07
7.	80386 Processor	07
8.	Current Era of Microprocessors	04
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**THE PROCESSOR: 8086**	**05 Hours**	**10%**
	Register Organisation of 8086, Architecture, Signal Descriptions, Physical Memory Organisation, General Bus Operation, I/O Addressing Capability, Special Purpose Activities, Minimum & Maximum Modes 8086 System and Timings.		
2.	**8086 INSTRUCTION SET AND ASSEMBLER DIRECTIVES**	**10 Hours**	**20%**
	Machine Language Instruction Formats, Addressing Modes, Instruction Set, Assembler Directives and Operators		

3.	**THE ART OF ASSEMBLY LANGUAGE PROGRAMMING WITH 8086**	**15 Hours**	**30%**
	A Few Machine Level Programs, Machine Coding the Programs, Programming with an Assembler, Assembly Language Example Programs		
4.	**SPECIAL ARCHITECTURAL FEATURES AND RELATED PROGRAMMING**	**10 Hours**	**20%**
	Introduction to Stack, Stack Structure, Interrupt and Interrupt Service Routines, Interrupt Cycle of 8086, Non-Maskable Interrupt, Maskable Interrupt, Interrupt Programming, MACROS, Timing and Delays.		
5.	**DMA CONTROLLER, MULTIMICROPROCESSOR SYSTEMS**	**10 Hours**	**20%**
	DMA Controller, Interconnection Topologies, Software Aspects of Multi-microprocessor Systems, Tightly Coupled and Loosely Coupled Systems		
6.	**80286 PROCESSOR**	**07 Hours**	**17%**
	Salient Features of 80286, Internal Architecture of 80286, Signal Description of 80286, Real Addressing Mode, Protected Virtual Address Mode (PVAM)		
7.	**80386 PROCESSOR**	**07 Hours**	**17%**
	Salient Features of 80386DX, Architectural and Signal Descriptions of 80386, Register Organisation of 80386, Addressing Modes, Data Types of 80386, Real Address Mode of 80386, Protected Mode of 80386, Segmentation.		
8.	**CURRENT ERA OF MICROPROCESSORS**	**04 Hours**	**06%**
	Core i3, i5, i7, xen Processor, multi core processors.		

Course Outcome (COs):

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

CO1	Recognize elements of digital logic circuit. Moving from design of single bit function to multibit function. (Flip flop, Logic Gates, Combinational Circuit). Design circuit for fixed function arithmetic function. Understand the notation of writing register transfer language.
CO2	Design and examine the different Arithmetic, Logic and Shift circuit & Design control unit of Arithmetic, Logic and Shift Circuit.
CO3	Differentiate and conceptualize instruction and arithmetic level parallelism & Identify and compare different methods for computer I/O mechanisms.
CO4	Demonstrate and evaluate computer arithmetic operations on integer and real

	numbers using hardwired algorithm.
CO5	Categorize memory organization and understand functioning of internal cache memory hardware.
CO6	Understand and differentiate n-way set associative memory.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. "Microprocessors and Interfacing Programming and Hardware", Douglas V Hall, McGraw-Hill Education India Pvt. Ltd.
  2. "Advanced Microprocessor and Peripherals –Architecture, Programming and Interfacing", A.K.Ray & K.M Bhurchandi, Tata Mc Graw Hill, 2006.
  3. "8086 Programming and Advance Processor Architecture", M. T. Savaliya, Wiley-India.

- **Reference book:**
  1. "Microcomputer systems: The 8086 / 8088 Family architecture, Programming and Design", Yncheng Liu,Glenn A.Gibson, second edition, Prentice Hall of India , 2006

2. The 8088 and 8086 Microprocessors: Programming, Interfacing and Applications, Walter A.Triebel, Avtar Singh, Prentice-Hall of India Pvt. Ltd.
3. "IBM PC Assembly language and programming", Peter Abel, fifth edition, Pearson education / Prentice Hall of India Pvt. Ltd, 2007.

❖ **Software:**

1. 8086 Simulator & Emulator

CE342: DESIGN & ANALYSIS OF ALGORITHMS

Credits and Hours:

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

Pre-requisite courses:

- Data Structure and Algorithms
- Programming language

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	To derive time and space complexity of algorithm.	03
2.	Analysis of Algorithm	06
3.	Greedy Algorithm	07
4.	Divide and Conquer Algorithm	07
5.	Dynamic Programming	08
6.	Exploring Graphs	04
7.	Backtracking & Branch & Bound	05
8.	String Matching and Introduction to NP- Completeness	05
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Basics of Algorithms and Mathematics**	**03 Hours**	**05%**
	What is an algorithm?, Performance Analysis, Model for Analysis- Random Access Machine (RAM), Primitive Operations, Time Complexity and Space Complexity		
2.	**Analysis of Algorithm**	**06 Hours**	**14%**
	The efficiency of algorithm, average and worst case analysis, elementary operation, Asymptotic Notation, Analysing control statement, Analysing Algorithm using Barometer, Solving recurrence Equation, Sorting Algorithm		

3.	**Greedy Algorithm**	**07 Hours**	**16%**
	General Characteristics of greedy algorithms, Problem solving using Greedy Algorithm Making change problem **Graphs:** Minimum Spanning trees (Kruskal's algorithm, Prim's algorithm , Graphs: Shortest paths; The Knapsack Problem; Job Scheduling Problem		
4.	**Divide and Conquer Algorithm**	**07 Hours**	**16%**
	Multiplying large Integers Problem, Binary Search Sorting (Merge Sort, Quick Sort), Matrix Multiplication, Exponential		
5.	**Dynamic Programming**	**08 Hours**	**18%**
	Introduction, The Principle of Optimality, Problem Solving using Dynamic Programming – Calculating the Binomial Coefficient, Making Change Problem, Assembly Line-Scheduling Knapsack Problem, Shortest Path Matrix Chain Multiplication, Longest Common Subsequence		
6.	**Exploring Graphs & Backtracking**	**04 Hours**	**09%**
	An introduction using graphs and games, Traversing Trees – Preconditioning Depth First Search- Undirected Graph; Directed Graph, Breath First Search, Applications of BFS & DFS		
7.	**Backtracking & Branch & Bound**	**05 Hours**	**12%**
	Backtracking –The Knapsack Problem; The Eight queens problem, General Template, Brach and Bound –The Assignment Problem; The Knapsack Problem, The min-max principle		
8.	**String Matching and Introduction to NP-Completeness**	**05 Hours**	**10%**
	The naïve string matching algorithm, The Rabin-Karp algorithm, The class P and NP Problem, Polynomial reduction, NP- Completeness Problem, NP-Hard problems		

Course Outcome (COs):

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

CO1	Analyse the asymptotic performance of algorithms.
CO2	Derive time and space complexity of different sorting algorithms and compare them to choose application specific efficient algorithm.
CO3	Understand and analyse the problem to apply design technique from divide and conquer, dynamic programming, backtracking, branch and bound techniques and understand how the choice of algorithm design methods impact the performance of programs.
CO4	Understand and apply various graph algorithms for finding shorted path and minimum spanning tree.
CO5	Synthesize efficient algorithms in common engineering design situations.
CO6	Understand the notations of P, NP, NP-Complete and NP-Hard.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald Rivest and Clifford Stein, MIT Press

❖ **Reference book:**

1. Fundamental of Algorithms by Gills Brassard, Paul Bratley, Pentice Hall of India.
2. Fundamental of Computer Algorithms by Ellis Horowitz, Sartazsahni and sanguthevar Rajasekarm, Computer Sci.P.
3. Design & Analysis of Algorithms by P H Dave & H B Dave, Pearson Education.

❖ **Web material:**

1. http://highered.mcgraw-hill.com/sites/0073523402/

CE343: SOFTWARE ENGINEERING

Credits and Hours:

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

Pre-requisite courses:

- N/A

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to Software and Software Engineering	04
2.	Agile Development	04
3.	Managing Software Project	05
4.	Requirement Analysis and Specification	04
5.	Software Design	05
6.	Software Coding &Testing	06
7.	Quality Assurance and Management	05
8.	Software Maintenance and Configuration Management	05
9.	Introduction to SaaS	03
10.	Advanced Topics in Software Engineering	04
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Introduction to Software and Software Engineering**	**04 Hours**	**09%**
	The Evolving Role of Software, Software: A Crisis on the Horizon and Software Myths, Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model,		

	Evolutionary Process Models, Agile Process Model, Component-Based Development, Process, Product and Process		
2.	**Agile Development**	**04 Hours**	**09%**
	Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools		
3.	**Managing Software Project**	**05 Hours**	**11%**
	Software Metrics (Process, Product and Project Metrics), Software Project Estimations, Software Project Planning (MS Project Tool), Project Scheduling & Tracking, Risk Analysis & Management(Risk Identification, Risk Projection, Risk Refinement ,Risk Mitigation)		
4.	**Requirement Analysis and Specification**	**04 Hours**	**09%**
	Understanding the Requirement, Requirement Modeling, Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, Requirement Engineering		
5.	**Software Design**	**10 Hours**	**20%**
	Design Concepts and Design Principal, Architectural Design, Component Level Design (Function Oriented Design, Object Oriented Design) (MS Visio Tool), User Interface Design, Web Application Design		
6.	**Software Coding & Testing**	**06 Hours**	**13%**
	Coding Standard and coding Guidelines, Code Review, Software Documentation, Testing Strategies, Testing Techniques and Test Case, Test Suites Design, Testing Conventional Applications, Testing Object Oriented Applications, Testing Web and Mobile Applications, Testing Tools (Win runner, Load runner)		
7.	**Quality Assurance and Management**	**05 Hours**	**11%**
	Quality Concepts and Software Quality Assurance, Software Reviews (Formal Technical Reviews), Software Reliability, The Quality Standards: ISO 9000, CMM, Six Sigma for SE, SQA Plan		

8.	**Software Maintenance and Configuration Management**	**05 Hours**	**11%**
	Types of Software Maintenance, Re-Engineering, Reverse Engineering, Forward Engineering, The SCM Process, Identification of Objects in the Software Configuration, Version Control and Change Control		
9.	**Introduction to SaaS**	**03 Hours**	**07%**
	Product Lifetime : Independent Product Vs. Continues Improvement, Service Oriented Architecture, Cloud Computing, SaaS Architecture		
10.	**Advanced Topics in Software Engineering**	**04 Hours**	**09%**
	Component-Based Software Engineering, Client/Server Software Engineering, Web Engineering, Reengineering, Computer-Aided Software Engineering, Software Process Improvement, Emerging Trends in software Engineering		

Course Outcome (COs):

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

CO1	Understand basics about software engineering principles, methods and practices and to analyze software requirement specification Prepare, SRS (Software Requirement Specification) document and SPMP (Software Project Management Plan) document.
CO2	Apply the concept of Functional Oriented and Object Oriented Approach for Software Design, To explain the software design strategies and to apply software measurement and metrics using Function point, Cyclomatic complexity and Healstead software science measures.
CO3	Recognize how to ensure the quality of software product, different quality standards and software review techniques.
CO4	Formulate problem by following Software Testing Life Cycle. Apply various testing techniques and test plan in. Design Manual Test cases for Software Project. Use automation testing tool students will be able test the software.
CO5	Able to understand modern Agile Development and Service Oriented Architecture Concept of Industry.
CO6	Analyze software risk with estimation parameters such as cost, effort,

	schedule/duration and understand the concepts of software maintenance, reengineering, reverse engineering, software configuration management.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Roger S. Pressman, Software engineering- A practitioner's Approach, McGraw-Hill International Editions
- **Reference book:**
  1. Engineering Software as a Service An Agile Software Approach, Armando Fox and David Patterson
  2. Ian Sommerville, Software engineering, Pearson education Asia
  3. Pankaj Jalote, An Integrated Approach to Software Engineering by, Springer
  4. Rajib Mall, Fundamentals of software Engineering, Prentice Hall of India.
  5. John M Nicolas, Project Management for Business, Engineering and Technology, Elsevier
- **Web material:**
  1. www.en.wikipedia.org/wiki/Software_engineering
  2. www.win.tue.nl
  3. www.rspa.com/spi

4. www.onesmartclick.com/engsineering/software-engineering.html
5. www.sei.cmu.edus
6. https://www.edx.org/school/uc-berkeleyx

CE344: COMPUTER NETWORKS

Credits and Hours:

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

Pre-requisite courses:

- Data Communication and Networking

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to Computer Networks	04
2.	Data Link Layer	08
3.	Medium Access Control Sub Layer	10
4.	Network Layer	12
5.	Transport Layer	08
6.	Application Layer	03
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Introduction to Computer Networks**	**04 Hours**	**09%**
	Uses of computer network, network hardware, network software, OSI model, TCP/IP model, Comparison of OSI and TCP/IP model, Example network		
2.	**Data Link Layer**	**08 Hours**	**18%**
	Design Issues, framing, error control, flow control, Error detection and correction, Elementary data link protocols, simplex, stop and wait, sliding window protocol, HDLC		
3.	**Medium Access Control Sub Layer**	**10 Hours**	**22%**
	The channel allocation problem, Multiple Access protocols: ALOHA, CSMA, Collision Free Protocols, Limited		

	Contention Protocols, Wavelength Division Multiple Access Protocols, Wireless LAN protocols; Ethernet: Traditional Ethernet, Switched Ethernet, Fast Ethernet, Gigabit Ethernet, IEEE 802.2: LLC, Data link layer switching		
4.	**Network Layer**	**12 Hours**	**27%**
	Implementation of connection oriented and connection less service, Comparison of virtual circuit and datagram subnets, Routing algorithms, Shortest path routing, Flooding, Distance vector routing, Link state routing, Hierarchical routing, Broadcast routing, Multicast routing, Routing for mobile host, Routing in ad hoc network, Congestion control algorithms principles, Prevention policies, Congestion control in virtual circuit subnets, Congestion control in datagram subnets, Load shedding, virtual circuit, Connectionless internetworking, Tunneling, Internetwork routing and fragmentation, The network layer in the internet: The IP protocol, IP addresses, Internet control protocol, OSPF, BGP.		
5.	**Transport Layer**	**08 Hours**	**18%**
	The transport service: Services provided to the upper layers, Transport service primitives, Socket elements of transport protocols addressing, Connection establishment, Connection release, Flow control, Multiplexing, Crash recovery the transport protocol: UDP, TCP.		
6.	**Application Layer**	**03 Hours**	**06%**
	DNS: The DNS name space, Resource records, Name servers, Electronic mail: Architecture and services, World Wide Web: Architectural overview, HTTP.		

Course Outcome (COs):

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

CO1	Analyze layered network architecture and passage of data over communication links
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 autonomous networks
CO4	Analyze flow and rate control algorithms between a sender and receiver in wide area 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	-	-	-	-	-	-	-	-	-

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 "-"

Recommended Study Material:

- **Text book:**
  1. Computer Network, Andrew S. Tanenbaum, Prentice Hall PTR
- **Reference book:**
  - Introduction to Data Communication and Networking by Behrouz Forouzan, McGraw Hill
  - Data and Computer Communications, William Stallings, Prentice Hall
- **Web material:**
  1. http://www.cisco.com
  2. http://compnetworking.about.com

❖ **Software:**

1. Wireshark
2. Cisco packet tracer
3. Network Simulator

CE371: ADVANCED JAVA PROGRAMMING (PE-I)

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	2	4	0	6	**4**
Marks	100	100	0	200	

Pre-requisite courses:

- Object oriented Language – Java
- DBMS
- Web Programming

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	JDBC Programming	05
2.	Servlet	06
3.	Java Server Pages	05
4.	Java Server Faces 2.0	03
5.	Hibernate	05
6.	JAVA web Frameworks: Spring MVC	06
	Total hours (Theory) :	30
	Total hours (Lab) :	60
	Total hours :	90

Detailed Syllabus:

1.	**JDBC Programming**	**05 Hours**	**11 %**
	The JDBC Connectivity Model, Database Programming: Connecting to the Database, Creating a SQL Query, Getting the Results, Updating Database Data Error Checking, SQLException Class, The SQLWarning Class, Statement Interface, ResultSet Interface, Updatable Result Sets, Executing SQL Queries, ResultSetMetaData, Executing SQL Updates, Transaction Management		

2.	**Servlet**	**06 Hours**	**13 %**
	Overview of Servlet, Servlet Life Cycle, HTTP Methods, Structure and Deployment descriptor, ServletContext and ServletConfig interface, Attributes in Servelt, Request Dispacher interface, The Filter API: Filter, FilterChain, Filter Config, Cookies and Session Management: Understanding state and session, Understanding Session Timeout and Session Tracking, URL Rewriting		
3.	**Java Server Pages**	**05 Hours**	**11%**
	JSP Overview: The Problem with Servlets, Life Cycle of JSP Page, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment, JSP Directives, JSP Action, JSP Implicit Objects, JSP Form Processing, JSP Session and Cookies Handling, JSP Session Tracking, JSP Database Access, JSP Standard Tag Libraries, JSP Custom Tag, JSP Expression Language, JSP Exception Handling, JSP XML Processing		
4.	**Java Server Faces 2.0**	**03 Hours**	**07%**
	Introduction to JSF, JSF request processing Life cycle, JSF Expression Language, JSF Standard Component, JSF Facelets Tag, JSF Convertor Tag, JSF Validation Tag, JSF Event Handling and Database Access, JSF Libraries: PrimeFaces		
5.	**Hibernate**	**05 Hours**	**11%**
	Overview of Hibernate, Hibernate Architecture, Hibernate Mapping Types, Hibernate O/R Mapping, Hibernate Annotation, Hibernate Query Language		
6.	**Java Web Frameworks: Spring MVC**	**06 Hours**	**13%**
	Overview of Spring, Spring Architecture, bean life cycle, XML Configuration on Spring, Aspect – oriented Spring Managing Database, Managing Transaction		

Course Outcome (COs):

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

CO1	Illustrate database access and details for managing information using the JDBC API

CO2	Design to map java classes and object associations to relational database tables with hibernate framework.
CO3	Demonstrate web application lifecycle using servlet, and JSP.
CO4	Identify the user interface control and build custom tag library using JSF
CO5	Develop real-world applications using java frameworks and understands about MVC architecture with its application.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. SCWCD, Matthew Scarpino, Hanumant Deshmukh, Jignesh Malavie, Manning publication
  2. Hibernate 2nd edition, Jeff Linwood and Dave Minter, Beginning Après publication
  3. Java Persistence with MyBatis 3, K. Siva Prasad Reddy, PACKT publication
  4. Spring in Action 3rd edition , Craig walls, Manning Publication
  5. Java Server Faces in Action, Kito D. Mann, Manning Publication

- **Reference book:**
  1. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley

2. Beginning JSP, JSF and Tomcat, Giulio Zambon, Apress
3. JSF2.0 CookBook, Anghel Leonard, PACKT publication

❖ **Web material:**

1. http://www.service-architecture.com/application-servers/articles/j2ee_web_site_architecture.html
2. http://www.oracle.com/technetwork/java/javaee/overview/index.html
3. http://www.roseindia.net/struts/hibernate-spring/index.shtml

❖ **Software:**

1. JDK 8+
2. NetBeans IDE
3. Eclipse
4. IntelliJ IDEA

CE372: ADVANCED PROGRAMMING USING .NET FRAMEWORK (PE-I)

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	2	4	0	6	**4**
Marks	100	100	00	200	

Pre-requisite courses:

- Basics of Object Oriented Concepts, XML, Database Management System

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to .NET Framework	03
2.	C# Programming	08
3.	ADO.NET	06
4.	ASP.NET	06
5.	Creating and Consuming Web Services	03
6.	Advanced in .NET	04
	Total hours (Theory) :	30
	Total hours (Lab) :	60
	Total hours :	90

Detailed Syllabus:

1.	**Introduction to .NET Framework**	**03 Hours**	**10%**
	.NET framework Architecture, MSIL, CLR, CLS, CTS, Garbage Collection, Side by Side executing and Versioning, Application Domain and Assemblies, Namespace and Based Class Library, The End to DLL Hell - Managed Execution.		
2.	C**# Programming**	**08 Hours**	**26%**
	Basics: Types, Variables, Methods, Operators, Branching, Expression, Statements, Structs, Array, Enums. OOP: Classes and Object, Inheritance and Polymorphism, Operator Overloading, Interface, Delegates and Event, Exception handling, Generics and Collections, Multithreading, String Manipulation and Reflection API,Windows Application:		

	Menus–Dialogs – ToolTips, Windows Form Control, GDI+, Visual Inheritance in C#.		
3.	**ADO.NET**	**06 Hours**	**20%**
	Introduces the basic ADO.NET objects and their roles, Connection-based programming to perform live record updates, insert and deletion, Interaction with stored procedures and parameterized command,ADO.NET Disconnected Approach: DataSet, DataTable, Data View and Data Adaptor		
4.	**ASP.NET**	**06 Hours**	**20%**
	Basics: Introduction to ASP.NET, Working with Web and HTML Controls, Using Rich Server Controls, Login controls, Overview of ASP.NET Validation Controls, Complex Validators, Themes and Master Pages, Controls: Standard, Data, Grid View, Data List, List View, Tree View, Repeater, List-Bound State Management: Preserving State in Web Applications and Page-Level State, Cookies, Session, Application, Global. asax file		
5.	**Creating and Consuming Web Services**	**03 Hours**	**10%**
	The Motivation for Web Services, Role of XML for creating SOAP based Web Service, Understanding SOAP, WSDL and UDDI,Creating and Consuming RESTful (Representational State Transfer) web services using WCF		
6.	**Advanced in .NET**	**04 Hours**	**14%**
	Introduction to Windows Presentation Foundation (WPF),Introduction to MVC Framework, Comparison between MVCs,Introduction to Entity Framework (ORM),Introduction to Microsoft Azure Services		

Course Outcome (COs):

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

CO1	Understand code solutions and compile C# projects within the .NET framework.
CO2	Design and develop professional console and window based .NET application
CO3	Demonstrate knowledge of object-oriented concepts Design user experience and

	functional requirements C#.NET application.
CO4	Construct classes, methods, and assessors, and instantiate objects. Understand and implement string manipulation, events and exception handling within .NET application environment.
CO5	Design and Implement Windows & Web Applications using Windows Forms, ,Web Forms ,Control Library, Advanced UI Programming & Data Binding concepts
CO6	Design and Implement database connectivity using ADO.NET in window based application.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- Text Books:
  1. Professional C# .Net, Christian Nagel, Wrox Publication
  2. ASP.NET Complete Reference, Matthew Macdonald and Robert Standefer, TMH.
  3. ASP.NET 4 Unleashed (Hardcover) by Stephen Walther, Kevin Scott Hoffman.
- Reference Books:
  1. C# The Basics, Vijay Mukhi, BPB Publications
  2. ADO.NET in a Nutshell By Bill Hamilton, Matthew MacDonald
  3. Beginning Entity Framework Core 2.0 Database Access from .NET, Authors: Rouleau, Derek J., Apress
  4. Pro ASP.NET MVC 5, Authors: Freeman, Adam, Apress

❖ **Web Materials:**

1. https://docs.microsoft.com/en-us/dotnet/
2. https://www.tutorialspoint.com/asp.net/index.htm
3. www.c-sharpcorner.com
4. www.csharp-station.com/Tutorial.aspx

CE373: MOBILE APPLICATION DEVELOPMENTS (PE-I)

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	2	4	-	6	**4**
Marks	100	100	-	200	

Pre-requisite courses:

- Basic Design concept with XML, Database management system.

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Getting an Overview of Android	2
2.	Working with the User Interface Using Views and View Groups	6
3.	Intents and Fragments in Android	4
4.	Database Connectivity	3
5.	Introduction to Xcode and InterfaceBuilder for iOS	3
6	Model Development with Swift	6
7	Intro to Scrollable Views, Tabs and Pages	3
8	Displaying and Persisting Data	3
	Total hours (Theory) :	30
	Total hours (Lab) :	60
	Total hours :	90

Detailed Syllabus:

1.	**Getting an Overview of Android**	**02 Hours**	**08%**
	Android OS Architecture, Introducing Development Framework, Dalvik Virtual Machine – DVM, Android Virtual Device and SDK Manager, Developing and Executing the First Android Application, Android Activities- Creating an Activity, Managing the Lifecycle of an Activity,		
2.	**Working with the User Interface Using Views and**	**06 Hours**	**18%**

	ViewGroups		
	Working with Views- Text, EditText, Button, Radio Button, CheckBox, ImageButton, ToggleButton, RatingBar, Working with View Groups- LinearLayout, RelativeLayout, ConstraintLayout, ScrollView, Table, Frame, Table with ActionBar, Binding Data with the AdapterView Class- ListView, Spinner, GallaryView, Creating Menus & Dialogs		
3.	**Intents and Fragments in Android**	**04 Hours**	**14%**
	Intent Objects, Intent Filters, Linking the Activities Using Intent, Obtaining Results from Intent, Passing Data Using an Intent Object, Fragments- Fragment Implementation, Finding Fragments, Adding, Removing, and Replacing Fragments		
4.	**Database Connectivity**	**03 Hours**	**09%**
	SQLite Database, SQLite Data Types, Cursors and Content Values, SQLite Open Helper, Adding, Updating and Deleting Content, XML & JSON Based Web Services.		
5.	**Introduction to Xcode and InterfaceBuilder for iOS**	**03 Hours**	**09%**
	Xcode Intro: Demo of a basic iOS App, StoryBoards, Source files, & wiring them together, The View hierarchy and view attributes		
6	**Model Development with Swift**	**06 Hours**	**18%**
	Swift language essentials: Arrays, Dictionaries, functions, Optionals, Control Flow, Structs Enums and Classes, Playgrounds, Elements of The Swift Foundation classes, CocoaTouch Foundation Framework, Simple connections to the User Interface		
7	**Intro to Scrollable Views, Tabs and Pages**	**03 Hours**	**14%**
	Frames and Bounds, ScrollViews, TableViews, CollectionViews and their controllers, Tabbed Views and their controllers, PageController		
8	**Displaying and Persisting Data**	**03 Hours**	**10%**
	Using the Table View, Application Preferences, Database Storage Using SQLite		

Course Outcome (COs):

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

CO1	Understand various technologies and business trends impacting mobile applications
CO2	Apply a deep knowledge of mobile device, features, architecture and android functionality.
CO3	Analyse and implement frameworks, database and design patterns in Mobile Applications
CO4	Create a small but realistic working mobile application using features such as data persistence and data communications
CO5	Create a mobile application using the Swift programming language.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Android Developer Tools Essentials by Mike Wolfson - O'Reilly Media Publications
  2. Christian Keur and Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide, 5th edition, 2015
- **Reference book:**
  1. Learn Java for Android Development, 2nd Edition - Jeff Friesen - Apress Publications

2. Suzanne Ginsburg, Designing the iPhone User Experience: A User-Centered Approach to Sketching and Prototyping iPhone Apps, Addison-Wesley Professional, 2010
3. Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 2nd edition, 2015.

❖ **Web material:**

1. http://www.youtube.com/watch?v=SUOWNXGRc6g&list=PL2F07DBCDCC01493A
2. Study Tutorial: https://developer.android.com/sdk/index.html
3. https://www.xamarin.com/forms
4. https://docs.microsoft.com/en-us/xamarin/
5. https://developer.apple.com/xcode/

❖ **Software:**

1. Android Studio
2. Flutter
3. Xcode

CE352: SOFTWARE GROUP PROJECT-III

Credits and Hours:

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

Pre-requisite courses:

- Programming Language, Software Engineering.

Outline of the Course:

- Student at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students will work together in a team (at most three) with any programming language.
- 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 of continuous 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.
  g. Project Synopsis
  h. Software Requirement Specification
  i. SPMP
  j. Final Project Report
  k. Project Setup file with Source code
  l. Project Presentation (PPT)

- A student has to produce some useful outcome by conducting experiments or project work.

Total hours (Theory) : 00

Total hours (Lab) : 60

Total hours : 60

Course Outcome (COs):

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

CO1	An ability to function effectively in teams to accomplish a common goal.
CO2	An ability to apply knowledge of computing and engineering to evaluate project requirements.
CO3	An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs.
CO4	An ability to analyze the local and global impact of computing on individuals, organizations, and society.
CO5	Write technical report and deliver presentation by applying different visualization tools and evaluation metrics.
CO6	An ability to communicate effectively with end user.

Course Articulation Matrix:

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

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines ,Journals & online course platforms of related topics
- **Web material:**
    1. www.sciencedirect.com
    2. www.elsevier.com
    3. https://www.udemy.com/
    4. https://www.udacity.com/
    5. https://nptel.ac.in/course.html
    6. https://www.futurelearn.com/
- **Software:**
    1. ASP.NET
    2. PYTHON/MATLAB
    3. PHP
    4. ANDROID/IOS

HS 131.02 A: COMMUNICATION AND SOFT SKILLS

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	--	02/01	--	30/15	**02**
Marks	--	100	--	100	

Pre-requisite courses:

- Communicative English

Objectives of the Course:

- To hone and sharpen Communication Skills of students
- To prepare globally and multi-culturally competent communicators and professionally compatible cadre of future professionals
- To equip and empower students to qualify and successfully clear all the phases of selection procedure for on and off campus interviews

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	An Introduction to Communication	06
2.	Cross-cultural Communication and Globalization	03
3.	Communication for Career Building	10
4.	Group Dynamics and Soft Skills	05
5.	Effective Presentation Strategies	04
6.	Contemporary Issues in Communication and Soft Skills	02
	Total hours (Theory) :	--
	Total hours (Practical) :	30
	Total hours :	30

Detailed Syllabus:

1.	**An Introduction to Communication**	**06 Hours**	**20%**
	Basics of Communication: Origin, Concept, Process, Levels, Principles and Barriers; Applications of Communication; Rhetoric in Professional Communication; Importance of Ethos, Logos, and Pathos in Communication		
2.	**Cross-cultural Communication and Globalization**	**03 Hours**	**10%**

	Basic Concepts: Culture, Globalization and Cross-cultural Communication; Social and People Skills; Communicating with People of Different Cultures; Conflicts in Cross-cultural Communication and Tactics / techniques to resolve them; Persuasive Communication		
3.	**Communication for Career Building**	**10 Hours**	**33%**
	Cover Letters and Resume; E-mail and Report; Types of Resume; Concept and Rationale of Group Discussion Skills and Aspects assessed in Group Discussion; Concept and Rationale of Personal Interview; Types of Personal Interview; Writing Statement of Purpose		
4.	**Group Dynamics and Soft Skills**	**05 Hours**	**17%**
	An Introduction to Group Dynamics and Soft Skills;Groups and their Structures; Roles and Functions of Members in Groups; Conflict Management; Aptitude and Attitude; Various Intelligences; Developing an Open Mindset		
5.	**Effective Presentation Strategies**	**04 Hours**	**14%**
	Designing Appealing Presentation; Audience Analysis and Supporting Material; Presentation Mechanics and Presentation Process; Managing Yourself during Q and A Session; Fundamentals of Persuasion		
6.	**Contemporary Issues in Communication and Soft Skills**	**02 Hours**	**06%**
	Trends and Practices in Communication, Case Studies		

Course Outcome (COs):

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

CO1	Gain thorough understanding and proficiency in various Professional Communication Skills.
CO2	Develop awareness and competence in cross-cultural communication in their personal, academic and professional environments.
CO3	Develop business writing and presentation skills to succeed in career.
CO4	Develop soft skills to stand out and take their career to the next level.
CO5	Develop various intelligences and open Mindset to function in multi-disciplinary and cross-cultural work environment.
CO6	Practice new trends in communication in multiple perspectives at personal, professional, and social level.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2

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

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 "-"

Recommended Study Material:

- **Text book:**

1. Koneru, A. Professional Communication, Tata McGrow Hill Education Private Limited
2. Disanza, J.R. &Legge, N. Business and Professional Communication, Pearson Education
3. Raman, M & Singh, P. Business Communication, Oxford University Press

- **Reference book:**

1. Disanza, J.R. &Legge, N. Business and Professional Communication, Pearson Education
2. Anandamurugan, A. Placement Interviews – Skills for Success, Tata McGrow Hill Education Private Limited

- **Web material:**

1. https://www.coursera.org/learn/careerdevelopment
2. https://www.futurelearn.com/courses/writing-applications
3. https://www.futurelearn.com/courses/workplace-englis

CE346: SUMMER INTERNSHIP-I

Credits and Hours:

Teaching Scheme	Project	Practical	Tutorial	Total	Credit
Hours	90	-	-	90	**3**
Marks	**150**	-	-	**150**	

Objectives of the Course:

- To get familiar with modern tools and technologies use in company/industry/organization
- To get involved in design, development and testing practices followed in the company/industry/organization
- To enhance their soft-skills, presentation skills, interpersonal skills, documentation skills and office etiquettes required to sustain in company/industry/organization environment
- To participate in teamwork and preferably as part of a multi-disciplinary team
- To make them aware about company/industry/organization best practices, processes and regulations.
- To make them more productive, consistent and punctual.

Outline of the Course:

1.	**Instructional Method and Pedagogy**
	• Summer internship shall be at least 90 hours during the summer vacation only. • Department/Institute will help students to find an appropriate company/industry/organization for the summer internship. • The student must fill up and get approved a Summer Internship Acceptance form by the company and provide it to the Coordinator of the department within the specified deadline. • Students shall commence the internship after the approval of the department Coordinator. Summer internships in research centers is also allowed. • During the entire period of internship, the student shall obey the rules and regulations of the company/industry/organization and those of the University. • Due to inevitable reasons, if the student will not able to attend the internship for few days with the permission of the supervisor, the department

	Coordinator should be informed via e-mail and these days should be compensated later.
	• The student shall submit two documents to the Coordinator for the evaluation of the summer internship:
	  • Summer Internship Report
	  • Summer Internship Assessment Form
	• Upon the completion of summer internship, a hard copy of "Summer Internship Report" must be submitted through the presentation to the Coordinator by the first day of the new term.
	• The report must outline the experience and observations gained through practical internship, in accordance with the required content and the format described in this guideline. Each report will be evaluated by a faculty member of the department on a satisfactory/unsatisfactory basis at the beginning of the semester.
	• If the evaluation of the report is unsatisfactory, it shall be returned to the student for revision and/or rewriting. If the revised report is still unsatisfactory the student shall be requested to repeat the summer internship.

2.	**Format of Summer Internship Report**
	The report shall comply with the summer internship program principles. Main headings are to be centered and written in capital boldface letters. Sub-titles shall be written in small letters and boldface. The typeface shall be Times New Roman font with 12pt. All the margins shall be 2.5cm. The report shall be submitted in printed form and filed. An electronic copy of the report shall be recorded in a CD and enclosed in the report. Each report shall be bound in a simple wire vinyl file and contain the following sections: • Cover Page • Page of Approval and Grading • Abstract page: An abstract gives the essence of the report (usually less than one page). Abstract is written after the report is completed. It must contain the purpose and scope of internship, the actual work done in the plant, and conclusions arrived at.

- TABLE OF CONTENTS (with the corresponding page numbers)
- LIST OF FIGURES AND TABLES (with the corresponding page numbers)
- DESCRIPTION OF THE COMPANY/INDUSTRY/ORGANISATION: Summarize the work type, administrative structure, number of employees (how many engineers, under which division, etc.), etc. Provide information regarding
    - Location and spread of the company
    - Number of employees, engineers, technicians, administrators in the company
    - Divisions of the company
    - Your group and division
    - Administrative tree (if available)
    - Main functions of the company
    - Customer profile and market share
- INTRODUCTION: In this section, give the purpose of the summer internship, reasons for choosing the location and company, and general information regarding the nature of work you carried out.
- PROBLEM STATEMENT: What is the problem you are solving, and what are the reasons and causes of this problem.
- SOLUTION: In this section, describe what you did and what you observed during the summer internship. It is very important that majority of what you write should be based on what you did and observed that truly belongs to the company/industry/organization.
- CONCLUSIONS: In the last section, summarize the summer internship activities. Present your observations, contributions and intellectual benefits. If this is your second summer internship, compare the first and second summer internships and your preferences.
- REFERENCES: List any source you have used in the document including books, articles and web sites in a consistent format.
- APPENDICES: If you have supplementary material (not appropriate for the main body of the report), you can place them here. These could be schematics, algorithms, drawings, etc. If the document is a datasheet and it can be easily

	accessed from the internet, then you can refer to it with the appropriate internet link and document number. In this manner you don't have to print it and waste tons of paper.	
	Total hours (Project) :	**90**
	Total hours :	**90**

Course Outcome (COs):

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

CO1	Ability to integrate existing and new technical knowledge for industrial application.
CO2	Executing work with team and teammates from other disciplines
CO3	Get practices and experience related to professional and ethical issues in the work environment
CO4	Experience of demonstrating the impact of the internship on their learning and professional development.
CO5	Understanding of lifelong learning processes through critical reflection of internship experiences.

Course Articulation Matrix:

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

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines & Journals of related topics
- **Web material:**
    1. www.ieeexplore.ieee.org
    2. www.sciencedirect.com
    3. www.elsevier.com
    4. http://spie.org/x576.xml
- **Software**
    1. ASP.NET
    2. PYTHON/MATLAB
    3. PHP
    4. ANDROID/IOS
    5. FLUTTER
    6. NODE/REACT NATIVE

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 6)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE347: INTERNALS OF OPERATING SYSTEM

Credits and Hours:

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

Pre-requisite courses:

- Operating System

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to the Kernel	03
2.	Kernel Memory Allocation	06
3.	Internal Representation of Files	10
4.	The Structure of Processes	06
5.	Process Control	06
6.	Memory Management Policies	04
7.	Interprocess Communication	03
8.	Distributed File Systems	02
9.	Introduction to System Administration	03
10.	Case Study: RTOS, Network Operating System	02
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Introduction to the Kernel**	**03 Hours**	**06%**
	Architecture of the Linux/Unix Operating System, Introduction to system concepts, kernel data structures, system administration		
2.	**Kernel Memory Allocation**	**06 Hours**	**11%**
	Introduction, Functional Requirements, Resource Map		

	Allocator, The McKusick-Karels Allocator, The buddy system, The SVR4 lazy buddy algorithm		
3.	**Internal Representation of Files**	**10 Hours**	**18%**
	Buffer headers, Structure of the Buffer Pool, Scenarios for Retrieval of a Buffer, Reading and Writing Disk Blocks, Inodes, Structure Of A Regular File, Directories, Conversion Of A Path Name To An Inode, Super Block, Node Assignment To A New File, Allocation Of Disk Blocks, Other File Types		
4.	**The Structure of Processes**	**06 Hours**	**15%**
	Process States and Transactions, Layout of System Memory, The Context of a Process, Saving the Context of a Process, Manipulation of the Process Address Space, Sleep		
5.	**Process Control**	**06 Hours**	**16%**
	Process Creation, Signals, Process Termination, Awaiting Process Termination, Invoking other Programs, The User ID of a Process, Changing the Size of a Process, The SHELL, System Boot and INIT process		
6.	**Memory Management Policies**	**04 Hours**	**08%**
	Swapping, Demand Paging, A Hybrid system with Swapping and Demand Paging		
7.	**Interprocess Communication**	**03 Hours**	**06%**
	Process Tracing, System V IPC, Network Communications, Sockets		
8.	**Distributed File Systems**	**02 Hours**	**05%**
	General Characteristics Of Distributed File System, Network File System, Remote File System		
9.	**Introduction to System Administration**	**03 Hours**	**10%**
	Basics of System Administration		
10.	**Case Study: RTOS, Network Operating System**	**02 Hours**	**05%**

Course Outcome (COs):

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

CO1	Describe an overview of Kernel architecture and outlining basic concepts of file subsystem and process, Discover various functionalities of Kernel and apply basic understanding of kernel in subsequent modules.
CO2	Compare and Contrast Various Kernel memory allocators based on evaluation criteria and Discuss memory management policies and test the performance on various paging algorithms.
CO3	Examine internal structure of file and develop lower level file system algorithms.
CO4	Interpret and formulate context of a process theoretically and programmatically, examine how one process traces and controls the execution of other process and implement system calls that control the process context.
CO5	Review the traditional methods by which process communicate with processes on other machines over a network and to Summarize the characteristics of Distributed File system to conclude strength and weakness of network file systems.
CO6	Describe basics of system administration in operating system and Apply various operating system concepts to understand case study of RTOS and Network Operating system.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. J.Maurice Bach, “The Design of the Unix Operating System”, Third Edition, Pearson Education, 2015
  2. Uresh Vahalia, “Unix Internals: The New Frontiers”, Pearson Education, 2008
  3. Michael Beck, Mirko Dziadzka, Claus Schroter, Dirk Verworner “Linux Kernel Programming” 3rd Edition , Pearson Education

- **Reference book:**
  1. Robert Love, “Linux Kernel Development”, 3rd Edition, Addison Wesley
  2. David Reilly and Michael Reilly “Java Network Programming and Distributed Computing” , Addison-Wesley
- **Web material:**
  1. Video lectures and Lecture Notes on "Operating Systems" by Prof. P.K. Biswas sir, IITKGP
- **Software:**
  1. Virtual box 6.0

CE348: INFORMATION SECURITY

Credits and Hours:

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

Pre-requisite courses:

- N/A

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction	02
2.	Traditional Symmetric-Key Ciphers	05
3.	Introduction to Modern Symmetric-Key Ciphers	04
4.	Data Encryption Standard (DES)	04
5.	Advanced Encryption Standard (AES)	04
6.	Encipherment Using Modern Symmetric-Key Ciphers	04
7.	Asymmetric-Key Cryptography	05
8.	Message Integrity and Message Authentication	04
9.	Cryptographic Hash Functions	04
10.	Digital Signature	04
11.	Entity Authentication	04
12.	Key Management	04
13.	Security at the Application Layer: PGP and S/MIME	04
14.	Security at the Transport Layer: SSL and TLS	04
15.	Security at the Network Layer: IPSec	04
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	90

Detailed Syllabus:

1.	**Introduction**	**02 Hours**	**06%**
	Security Goals, Attacks, Services and Mechanism and Techniques.		
2.	**Traditional Symmetric-Key Ciphers**	**05 Hours**	**10%**
	Introduction, Substitution Cipher, Transposition Cipher, Stream and Block Cipher		
3.	**Introduction to Modern Symmetric-Key Ciphers**	**04 Hours**	**06%**
	Modern Block Cipher, Modern Stream Cipher		
4.	**Data Encryption Standard (DES)**	**04 Hours**	**05%**
	Introduction, DES structure and Analysis, Multiple DES		
5.	**Advanced Encryption Standard (AES)**	**04 Hours**	**05%**
	Introduction, AES structure and Analysis		
6.	**Encipherment Using Modern Symmetric-Key Ciphers**	**04 Hours**	**06%**
	Use of Modern Block Ciphers, Use of Modern Stream Ciphers		
7.	**Asymmetric-Key Cryptography**	**05 Hours**	**10%**
	Introduction, RSA Cryptosystem, RABIN Cryptosystem, ELGAMAL Cryptosystem		
8.	**Message Integrity and Message Authentication**	**04 Hours**	**08%**
	Message Integrity, Message Authentication		
9.	**Cryptographic Hash Functions**	**04 Hours**	**08%**
	Introduction, SHA-512,MD5		
10.	**Digital Signature**	**04 Hours**	**06%**
	Services,RSA Digital Signature Scheme,ELGamal Digital Signature Scheme		
11.	**Entity Authentication**	**04 Hours**	**04%**
	Passwords, Challenge –Response,Zero –Knowledge		
12.	**Key Management**	**04 Hours**	**08%**
	Symmetric-Key Distribution,KERBEROS		
13.	**Security at the Application Layer: PGP and S/MIME**	**04 Hours**	**06%**
	E-Mail, PGP,S/MIME		
14.	**Security at the Transport Layer: SSL and TLS**	**04 Hours**	**06%**

	SSL Architecture,TLS cipher suite		
15.	**Security at the Network Layer: IPSec**	**04 Hours**	**06%**
	Introduction to IPSec		

Course Outcome (COs):

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

CO1	Define various security goal and understand the security policies such as the CIA triad of Confidentiality, Integrity and Availability.
CO2	Classify various forms of security attacks, where they arise, and appropriate tools or mechanism to quantify them.
CO3	Illustrate a basic understanding of cryptography, how it has evolved, and evaluate symmetric key encryption techniques used today.
CO4	Distinguish modern symmetric encryption standard, key distribution scenario and analyse effectiveness in todays' environment.
CO5	Evaluate Asymmetric key encryption techniques, key distribution scenario and calculate public and private components of asymmetric key encryption techniques.
CO6	Develop message integrity and message authentication of message digest.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Cryptography and Network Security, Behrouz A. Forouzan, McGraw-Hill Companies.
- **Reference book:**
  1. Cryptography And Network Principles And Practice, William Stallings, Prentice Hall, Pearson Education Asia.
  2. Cryptography & Network Security, Atul Kahate, The McGraw-Hill Companies.
  3. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations by Gene Kim, Jez Humble, Patrick Debois, and John Willis
- **Web material:**
  1. http://people.csail.mit.edu/rivest/crypto-security.html.
  2. http://www.cryptix.org/
  3. http://www.cryptocd.org/
  4. http://www.cryptopp.com/
  5. http://www.freetechbooks.com/information-security-f52.html
- **Software:**
  1. Nmap
  2. Wireshark

CE349: THEORY OF COMPUTATION

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	0	-	3	**3**
Marks	100	0	-	100	

Pre-requisite courses:

- Mathematics, Data Structures and Algorithms, Design and Analysis of Algorithm (Computational Complexity)

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1	Introduction	03
2	Mathematical Terms and Theory	05
3	Regular Grammar & Languages, Regular Expression, Finite Automata	14
4	Context Free Grammar & Languages, Push down Automata	13
5	Turing Machine, Recursively Enumerable Languages	08
6	Decidable & Undecidable Problems	02
	Total hours (Theory) :	45
	Total hours (Lab) :	00
	Total hours :	45

Detailed Syllabus:

1.	**Introduction**	**03 Hours**	**06 %**
	Alphabet, String, Language, Formal Grammar, Chomsky Hierarchy, Introduction to Automata		
2.	**Mathematical Terms and Theory**	**05 Hours**	**10 %**
	Mathematical Inductions, Recursive Definitions		
3.	**Regular Grammar & Languages, Regular Expression, Finite Automata**	**14 Hours**	**32 %**

	Regular Language, Regular Expressions, Applications, Chomsky Hierarchy, Finite Automata, Nondeterministic Finite Automata, Kleen's Theorem, Automata with Output (Moore Machine, Mealy Machine), Properties of Regular Languages (Pumping Lemma, Closure Property, Decision Algorithm)		
4.	**Context Free Grammar & Languages, Push down Automata**	**13 Hours**	**30 %**
	The Chomsky, Notion of Grammars and Languages Generated by Grammars, CFG, CFL, Regular Language and Regular Grammar, Derivation Tree and Ambiguity, BNF, CNF, GNF, CFL properties (Pumping Lemma, Closure Property, Decision Algorithm), Intersections and Complements of CFL, Non-CFL, Definition, DPDA, NPDA, Equivalence of CFG and PDA		
5.	**Turing Machine, Recursively Enumerable Languages**	**10 Hours**	**20%**
	Definition, Model of Computation, Combining TM, Variations of TM, Non Deterministic TM, Universal TM, Recursively Enumerable and Recursive, Enumerable Languages, Context sensitive languages		
6.	**Decidable & Undecidable Problems**	**02 Hours**	**04 %**
	Tractable and Intractable Problems, Complexity Classes, Tractable and Possibly Intractable Problems, P and NP Completeness, Countability		

Course Outcome (COs):

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

CO1	Apply basic concepts of theory of computation in the computer field in order to solve computational problems.
CO2	Construct algorithms for different problems and argue formally about correctness on different restricted machine models of computation.
CO3	Analyze and design finite automata, pushdown automata and Turing machine for formal languages.

CO4	Apply rigorously formal mathematical methods to prove properties of languages, grammars and automata.
CO5	Identify limitations of some computational models and possible solutions.
CO6	Design context free grammars for formal languages.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Introduction to Languages and Theory of Computation, John C. Martin, TMH

- **Reference book:**
  1. An introduction to automata theory and formal languages, Adesh K. Pandey, S. K. Kataria & Sons
  2. Introduction to computer theory, Deniel I. Cohen, John Wiley & Sons Inc
  3. Computation: Finite and Infinite, Marvin L. Minsky, Prentice-Hall
  4. "An introduction to Formal Languages and Automata", Peter Linz, 6th edition, Jones & Bartlett Learning
  5. "Introduction to the Theory of Computation", Michael Sipser, 3rd edition, Cengage Learning.

❖ **Web material:**

1. https://www.youtube.com/playlist?list=PLEbnTDJUr_IdM___FmDFBJBz0zCsOFxfK
2. http://nptel.ac.in/courses/106103070/
3. http://nptel.ac.in/courses/106104028/
4. http://nptel.ac.in/courses/106106049/
5. https://www.youtube.com/watch?v=4GLC-s0PQLY

CE350: DATAWAREHOUSING & DATA MINING

Credits and Hours:

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

Pre-requisite courses:

- A course in database systems is recommended, as is a basic course on algorithms and data structures.

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction to Data Warehousing	06
2.	Concepts and techniques in Data Warehousing	06
3.	Introduction to data mining (DM)	06
4.	Data Pre-processing	08
5.	Concept Description & Association Rule Mining	08
6.	Classification and Prediction	07
7.	Advance topics	04
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Overview and concepts Data Warehousing**	**06 Hours**	**14%**
	What is data warehousing - The building Blocks, Defining Features – Data warehouses and data marts, Overview of the components, Metadata in the data warehouse, Need for data warehousing, Basic elements of data warehousing, Trends in data warehousing.		
2.	**OLAP in Data Warehouse**	**06 Hours**	**14%**
	OLAP (Online analytical processing) definitions, Difference		

	between OLAP and OLTP, Dimensional analysis - What are cubes?, Drill-down and roll-up - slice and dice or rotation, OLAP models, ROLAP versus MOLAP, defining schemas: Stars, snowflakes and fact constellations		
3.	**Introduction to data mining (DM)**	**06 Hours**	**14%**
	DM Functionalities, Classification of DM Systems, Issues in DM – KDD Process		
4.	**Data Pre-processing**	**08 Hours**	**18%**
	Why to pre-process data?, Data cleaning: Missing Values, Noisy Data, Data Integration and transformation, Data Reduction: Data cube aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Data Mining Primitives, Languages and System Architectures: Task relevant data, Kind of Knowledge to be mined, Discretization and Concept Hierarchy.		
5.	**Concept Description and Association Rule Mining**	**08 Hours**	**18%**
	What is concept description? Data Generalization and summarization-based characterization, Attribute relevance class comparisons Association Rule Mining, Market basket analysis- basic concepts, Finding frequent item sets: Apriori algorithm - generating rules		
6.	**Classification and Prediction**	**07 Hours**	**16%**
	What is classification and prediction?, Issues Classification using Decision trees, Linear and nonlinear regression, Introduction of tools such as DBMiner /WEKA DM Tools		
7.	**Advance topics**	**04 Hours**	**06%**
	Introduction of clustering, Spatial mining, Web mining, Text mining, Big Data: Introduction to Big Data and Hadoop, What is Big Data and what are the challenges to process Big Data, What technologies supports Big Data, Hadoop introduction and its history, Hadoop vs RDBMS, Introduction of Hadoop Distributed File System, Introduction of Map-Reduce		

Course Outcome (COs):

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

CO1	Interpret the contribution of data warehousing and data mining to the decision-support level of organizations.
CO2	Apply pre-processing statistical methods for any given raw data.
CO3	Design and Evaluate different dimensional modelling used for OLAP.
CO4	Categorize and differentiate between situations for applying different data-mining techniques: frequent pattern mining, association, correlation, classification, prediction, cluster, and outlier analysis.
CO5	Evaluate the performance of different data-mining models/algorithms with respect to their accuracy.
CO6	Conceptualise a data mining solution to a practical problem.

Course Articulation Matrix:

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

Recommended Study Material:

- Text Books:
  1. "Data Mining Concepts and Techniques", J. Han, M. Kamber, Morgan Kaufmann
  2. "Data Warehousing Fundamentals", Paulraj Ponnian, John Willey.

❖ **Reference Books:**

1. "Data mining: Concepts, models, methods and algorithms, M. Kantardzic, John Wiley &Sons Inc.
2. "Data Mining: Introductory and Advanced Topics", M. Dunham, Pearson Education.
3. "Data Mining", Pieter Adriaans, Dolf Zantinge , Pearson Education Asia

❖ **Web Materials:**

1. http://www.dataminingblog.com
2. http://www.kdnuggest.com

CE353: SOFTWARE GROUP PROJECT -IV

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	0	4	-	4	**4**
Marks	0	200	-	200	

Pre-requisite courses:

- Programming Knowledge

Outline of the Course:

- Student at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students will work together in a team (at most three) with any programming language.
- 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 a 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 of continuous 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
  e. Project Setup file with Source code
  f. Project Presentation (PPT)

- A student has to produce some useful outcome by conducting experiments or project work.

Total hours (Theory) : 00

Total hours (Lab) : 60

Total hours : 60

Course Outcome (COs):

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

CO1	Prepare team formation strategies and stages leading to the development of high performing, self-managing teams.
CO2	Identify and define the computing requirements of a problem to propose its appropriate solution.
CO3	Correlate knowledge of different subjects and apply theoretical knowledge to implement project for identified problem.
CO4	Apply engineering and management principles to achieve project goal.
CO5	Write technical report and deliver presentation by applying different visualization tools and evaluation metrics.
CO6	An ability to communicate effectively with a range of audiences.
CO7	Recognition of the need for and an ability to engage in continuing professional development.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	2	-	-	-	-	-	-	3	-	-	1	-	2
CO2	-	3	1	2	-	-	-	-	-	-	-	-	2	-
CO3	3	-	2	-	1	-	-	-	-	-	-	-	2	-
CO4	3	-	-	2	-	-	1	-	-	-	-	-	3	-
CO5	-	-	-	-	2	-	-	-	-	1	3	-	-	2
CO6	-	-	-	-	-	2	-	-	-	3	-	1	-	2
CO7	-	-	-	-	-	1	-	2	-	-	-	3	-	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 "-"

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines & Journals of related topics
- **Web material:**
    1. www.ieeexplore.ieee.org
    2. www.sciencedirect.com
    3. www.elsevier.com
    4. http://spie.org/x576.xml
- **Software:**
    1. ASP.NET
    2. PYTHON/MATLAB
    3. PHP
    4. ANDROID/IOS

CE374: Service Oriented Computing (PE-II)

Credits and Hours:

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

Pre-requisite courses:

- N/A

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
16.	Overview of SOA	03
17.	SOA — Architecture Fundamentals	06
18.	Web Services and Contemporary SOA (Part I: Activity Management and Composition)	06
19.	Web Services and Contemporary SOA (Part II:	08
20.	Building SOA (Planning and Analysis)	06
21.	Building SOA (Technology and Design)	10
22.	Fundamental WS-* Extensions	06
	Total hours (Theory) :	**45**
	Total hours (Lab) :	**30**
	Total hours :	**75**

Detailed Syllabus:

1.	**Overview of SOA**	**03 Hours**	**07%**
	The Promise of SOA, The Challenges of SOA: Reuse, Efficiency in Development, Integration of Applications and Data, Agility, Flexibility, and Alignment.Meeting the Challenge Reference Architecture Common Semantics Governance Business Process Modelling Design-Time		

	Service Discovery Model-Based Development		
2.	**SOA — Architecture Fundamentals**	**06 Hours**	**13%**
	Architectural Styles, Architectural Principles and Practices, SOA and Other Architectures, Enterprise Architecture, Software Architecture, EA, 4+1, and Services, What Is a Service?, Service Characteristics: Service, Granularity, Service Dimensions, Loose Coupling Is King: Location Transparency, Interface and Implementation, Data, Versioning, Interoperability and Platform Independence, Usage, Assumptions, and Knowledge, Common Service Patterns, Service Types and Purpose, SOA Reference Architecture.		
3.	**Web Services and Contemporary SOA (Part I: Activity Management and Composition)**	**06 Hours**	**13%**
	Message exchange patterns, Service activity, Coordination, Atomic transactions, Business activities, Orchestration		
4.	**Web Services and Contemporary SOA (Part II:Advanced Messaging, Metadata, and Security**	**08 Hours**	**18%**
	Addressing, Reliable messaging, Correlation, Policies, Metadata exchange, Security, Notification and eventing		
5.	**Building SOA (Planning and Analysis)**	**06 Hours**	**13%**
	SOA Delivery Strategies: SOA delivery lifecycle phases, The top-down strategy, The bottom-up strategy, Service Modeling: Service modeling guidelines, Classifying service model logic.		
6.	**Building SOA (Technology and Design)**	**10 Hours**	**23%**
	WSDL-related XML Schema language basics, WSDL language basics, SOAP language basics, Composition Guidelines: Steps to composing SOA,Considerations for choosing service layers, Considerationsfor positioning core SOA standardsService Design: Service design overview,		

	Entity-centricbusiness service design (a step-by-step process),Application service design, Task-centric business servicedesign Business Process Design: WS-BPEL language basics, WS-Coordination overview.		
7.	**Fundamental WS-* Extensions**	**06 Hours**	**13%**
	WS-Addressing, WS-ReliableMessaging, WS-Policy, WS-MetadataExchange, WS-SecuritySOA Platforms: SOA support in J2EE, SOA support in.NET.		

Course Outcome (COs):

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

CO1	Define various promise and challengesof service oriented architecture.
CO2	Classify various forms of architectures, integration and interoperability of architecture principles.
CO3	Illustrate a basic understanding of activity management, how it has evolved, and evaluate composition and message exchanging.
CO4	Classifying delivery life cycle phase and analyseservice model logic.
CO5	Evaluate web services using SOAP, WSDLand create application service design.
CO6	Explore platforms such as J2EE and .Net for reliable messaging, security and policies for efficient service discovery.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Mike Rosen, Boris Lublinsky, Kevin T. Smith, Marc J. Balcer "Applied SOA:Service-Oriented Architectureand Design Strategies" WileyPublishing, Inc, 2008
  2. Thomas Erl, "Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall Publication, 2008

- **Reference book:**
  1. Norbert Bieberstein, Sanjay Bose, Marc Fiammante, Keith Jones, Rawn Shah, "Service-Oriented Architecture Compass: Business Value, Planning, andEnterprise Roadmap", IBM Press Publication, 2005.
  2. Thomas Erl, "Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services", Prentice Hall Publication, 2004.
  3. SanjivaWeerawarana, Francisco Curbera, Frank Leymann, Tony Storey, DonaldF.Ferguson, "Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS- Addressing, WS-BPEL, WS-Reliable Messaging, and More", Prentice HallPublication, 2005.
  4. Eric Newcomer, Greg Lomow, "Understanding SOA with Web Services",Addison Wesley Publication, 2004.
  5. ERL, T.; Service-oriented architecture : a field guide to integrating XML and Web services, Prentice Hall PTR, 2004, ISBN- 0131428985.

- **Web material:**
  1. https://www.ibm.com/developerworks/library/ws-top10/
  2. http://www.oracle.com/technetwork/articles/javase/soa-142870.html
  3. https://docs.oracle.com/javaee/6/tutorial/doc/gijti.html.

CE375: DIGITAL IMAGE PROCESSING (PE-II)

Credits and Hours:

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

Pre-requisite courses:

- Computer Graphics

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Digital Image Fundamentals	04
2.	Spatial Domain Image Enhancement Techniques	06
3.	Frequency Domain Image Enhancement Techniques	05
4.	Image Restoration	04
5.	Colour Image Processing	02
6.	Image Compression	03
7.	Image Segmentation	06
8.	Morphological Image Processing	06
9.	Representation & Description	03
10.	Applications of Image Processing	06
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Digital Image Fundamentals**	**04 Hours**	**10%**
	Image Basics, Elements of visual perception, Human Eye Structure, Why digital images, The digital camera, Data types and 2d representation of digital images, Application fields of Image Processing, Image Acquisition Techniques, Single Sensor, Strip, Array Sensor, Image sampling and quantization, Discrete sampling model Quantization, Noise processes, Spatial & Grey level resolution, Image attributes, Image		

	Types, Basic relationship between pixels Neighbourhood Adjacency and Distance measures		
2.	**Spatial Domain Image Enhancement Techniques**	**06 Hours**	**13%**
	Intensity based transforms- power law, log ,image negative, Histogram based transforms- Histogram processing, Equalization, Local Enhancement, Specification, Image Averaging, Subtraction, AND-OR-NOT Operations between images, Smoothing Filters- Linear Filters, Order Statistic Filters, Sharpening Filters-Laplacian Filter , Unsharp masking , high boost filtering		
3.	**Frequency Domain Image Enhancement Techniques**	**05 Hours**	**10%**
	Introduction to signal-frequency concept, Discrete Fourier Transform and Inverse Transform, properties of Fourier Transform ,1D Fourier Transform, 2D Fourier Transform, Frequency Filtering Concepts, Smoothing Filters(Low pass Filters)- Ideal low-pass, Butterworth low-pass, Gaussian low-pass filters, Sharpening Filters(High-Pass filters)- Ideal , Butterworth, Gaussian, Laplacian Filter in Frequency domain, Unsharp masking , high boost filtering ,Homomorphic Filtering , Convolution & Correlation Theorem , Fast Fourier Transform		
4.	**Image Restoration**	**04 Hours**	**10%**
	Image Degradation-Restoration Model, Noise Models, Noise probability density functions, Noise estimation parameters, Spatial Domain Restoration- Mean filters – Arithmetic mean, Geometric mean, Harmonic mean Order Statistics Filters- Median filter, Max & min filter, Midpoint filter, Alpha trimmed filter, Adaptive filters, Adaptive local noise filter, Adaptive median filter, Frequency Domain Restoration Techniques- Band reject filter, Band-pass filter, Notch filter, Optimum Notch filter, Estimation of Degradation function-By experiment, By Modeling , Inverse Filtering, Wiener Filtering, Constrained Least Square Filtering, Geometric Mean filter		

5.	**Colour Image Processing**	**02 Hours**	**03%**
	Colour fundamentals , Colour Models, HIS, YIQ, RGB, CMYK, CIE Lab, XYZ, Intensity Slicing, Grey level to Colour Transform, Tone & Colour Correction, Histogram Processing , Smoothing, Sharpening, Segmentation		
6.	**Image Compression**	**03 Hours**	**10%**
	Fundamental to Data Compression, Information Theory, Rate Distortion Theory, Redundancy- Coding, Inter-pixel, psycho-visual redundancy, Fidelity, Image Compression Models, Lossless Compression theorem- Huffman, Arithmetic Coding , LZW, Bit-Plane Coding, Run Length Encoding, Contour Coding, Lossless predictive Coding , Lossy predictive Coding- Transform Coding, DCT, KLT, Wavelet, JPEG Compression, JPEG2000		
7.	**Image Segmentation**	**06 Hours**	**13%**
	Detection of Discontinuities, Point Detection, Line Detection, Edge Detection, Edge Linking and Boundary Detection, Local Processing, Global Processing via the Hough Transform, Global Processing via Graph Theoretic Techniques, Thresholding, Foundation, The Role of Illumination, Basic Global Thresholding, Basic Adaptive Thresholding, Optimal Global and Adaptive Thresholding, Use of Boundary Characteristics for Histogram Improvement and Local Thresholding, Thresholds Based on Several Variables Region, Based Segmentation, Basic Formulation, Region Growing, Region Splitting and Merging, Segmentation by Morphological Watersheds, Basic Concepts, Dam Construction, Watershed Segmentation Algorithm, The Use of Markers		
8.	**Morphological Image Processing**	**06 Hours**	**13%**
	Preliminaries: Some Basic Concepts from Set Theory, Logic Operations Involving Binary Images, Dilation and Erosion, Opening and		

	Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms: Boundary Extraction-, Region Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening, Skeletons, Pruning		
9.	**Representation & Description**	**03 Hours**	**07%**
	Chain Code, Polygonal Approximation, Signatures, Boundary segments, Skeletons, Shape numbers, Fourier Descriptors, Statistical moments, Regional Descriptors: Topological, Texture, Moments of 2D functions, Using Principle Components		
10.	**Applications of Image Processing**	**06 Hours**	**14%**
	Case Study on the following applications: Digital Water Marking, Biometric authentication (face, fingerprint, signature recognition), Vehicle number plate detection and recognition, Content Based Image Retrieval, Text Compression. Image Mining, Image Fusion.		

Course Outcome (COs):

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

CO1	Examine, explain and analyse different types of images, general terminology and fundamental concepts of digital image processing.
CO2	Interpret, compare and evaluate the techniques for image enhancement, restoration, segmentation, morphological operation, representation and description.
CO3	Interpret different colour models and apply different image processing algorithms on colour images.
CO4	Categorize and analyse various compression techniques.
CO5	Understand real time problems of image processing filed and apply image processing knowledge to solve those problem.

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	-	2	2	2	-	-	-	-	-	-	-	-	1
CO3	3	-	2	2	2	-	-	-	-	-	-	-	-	1
CO4	3	-	2	2	2	-	-	-	-	-	-	1		1
CO5	3	3	3	3	3	3	2	-	-	-	-	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 "-"

Recommended Study Material:

- **Text book:**
  1. "Digital Image Processing", Rafael C Gonzalez, Richard E Woods, 2nd Edition, Pearson Education 2003.
  2. "Digital Image Processing", W. Pratt, Wiley Publication, Fourth Edition, 2013.

- **Reference book:**
  1. "Fundamentals of Digital Image Processing", A.K. Jain, PHI, New Delhi (1995).
  2. "Digital Image Processing Using MATLAB", Gonzalez/Woods/Eddins , 2nd edition
  3. "Digital Image Processing and Computer Vision". Milan Sonka, Thomson publication, Second Edition.
  4. "Digital Image Processing", S.Jayaraman, S Esakkirajan and T Veerakumar, McGraw Hill Education (India) Private Limited, New Delhi, 2009
  5. "Digital Image Processing", S.Sridhar, Oxford University Press, New Delhi, 2011.

❖ **Web material:**

1. http://www.imageprocessingplace.com/

❖ **Software:**

1. MATLAB
2. OpenCV

CE376: PROGRAMMING IN PYTHON(PE-II)

Credits and Hours:

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

Pre-requisite courses:

- High level language (C/C++/Java)
- Web Programming

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Basics of Python	01
2.	Data Structures: Lists, Tuples, Dictionaries and Strings	03
3.	Control structures and Function	03
4.	Modules and Scoping Rules	04
5.	Exceptions Handling	04
6.	Magic Methods, Properties, and Iterators	05
7.	Object Oriented Programming	06
8.	Regular Expression and File Handling	04
9.	Django web Framework	05
10.	Working with Django Templates	05
11.	Working with Django Form	05
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Basics of Python**	**01 Hours**	**02 %**
	Using the Python Interpreter, Variables, Identifiers and Keywords, Numbers and Expressions		

2.	**Data Structures: List, Tuples, Dictionaries and Strings**	**03 Hours**	**07 %**
	Common Sequence Operations: Indexing, Slicing, Adding Sequences, Multiplication, Membership, Length, Minimum, and Maximum, Using Lists as Stacks, Using Lists as Queues, List Comprehensions, Nested List Comprehensions, The del statement, Tuples and Sequences, Sets, Dictionaries, Comparing Sequences and Other Types, Basic String Operations		
3.	**Control Structures and Functions**	**03 Hours**	**07%**
	Conditional Branching: if Statements, break and continue Statements, and else Clauses on Loops, pass Statements Loops: while Loops, for Loops, Defining Functions, More on Defining Functions: Default Argument Values, Keyword Arguments, Arbitrary Argument Lists, Unpacking Argument Lists, Lambda Expressions, Documentation Strings, Function Annotations		
4.	**Modules and Scoping Rules**	**04 Hours**	**09%**
	Executing modules as scripts, The Module Search Path, "Compiled" Python files, Packages: Importing * From a Package, Intra-package References, Packages in Multiple Directories		
5.	**Exception Handling**	**04 Hours**	**09%**
	Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Predefined Clean-up Actions		
6.	**Magic Methods, Properties and Iterators**	**05 Hours**	**11%**
	Constructors, Item Access: The Basic Sequence and Mapping Protocol, Properties: The property Function, Static Methods and Class Methods, __getattr__, __setattr__, and Friends, Iterators, Generators, Generator Expressions		
7.	**Object Oriented Programming**	**06 Hours**	**13%**
	Python Scopes and Namespaces, Class Definition, Class Objects, Instance Objects, Method Objects, Class and Instance Variables, Inheritance, Multiple Inheritance, Private Variables, Polymorphism, Using Properties to Control Attribute Access, Creating Complete Fully Integrated Data Types		

8.	**Regular Expression and File Handling**	**04 Hours**	**09%**
	What is a regular expression?, Regular expressions with special characters, Regular expressions and raw strings, Extracting matched text from strings, Substituting text with regular expressions, Writing and Reading Binary Data, Writing and Parsing Text Files, Iterating over File Contents, Writing and Parsing XML Files, Random Access Binary Files		
9.	**Introduction to Django Web Framework**	**05 Hours**	**11%**
	MVC Architecture, Object, Relational Mapping architecture, Django settings: Routing in Django, Views and URL confs		
10	**Working with Django Templates**	**05 Hours**	**11%**
	Injecting the data from the view to the template, Creating dynamic templates, Integrating variables in templates, Filters, Creating DRY URLs, Extending the templates, Using static files in templates.		
11	**Working with Django Model and Form**	**05 Hours**	**11%**
	Model: Databases and Django, Migrations with South, Creating simple models, The relationship between the models, Extending models, The admin module, Model's Data with Query sets, Forms: Adding a developer with/without using Django forms, The form based on a model, Advanced usage of Django forms		

Course Outcome (COs):

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

CO1	Interpret the fundamental python syntax, semantics and fluent in the use of python control flow statements. Express proficiency in the handling of strings and functions.
CO2	Determine the methods to create and manipulate python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
CO3	Identify the commonly used operations involving file systems and regular expressions.
CO4	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python along with magic methods.
CO5	Develop real-world applications using frameworks and understands about

	database with its application and forms.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Magnus Lie Hetland, "Beginning Python From Novice to Professional", Third Edition, Apress,2017
  2. Nigel George, "Mastering Django: Core" Packt Publishing, 2016

- **Reference book:**
  1. David Beazley, Brian K. Jones, "Python Cookbook", 3rd edition, OREILLY,2016
  2. Brett Slatkin, "Effective Python: 59 Specific Ways to Write Better Python", Novatec, 2016
  3. Allen Downey, "Think Python: How to Think Like a Computer Scientist", Green Tea Press,2015
  4. Mark Lutz "Learning Python", 4th Edition, O'REILLY, 2016
  5. Arun Ravindran, Aidas Bendoraitis, Samuel Dauzon, "Django: Web Development with Python",Packt Publishing, 2016

❖ **Web material:**

1. https://www.python.org/
2. http://www.diveintopython3.net/
3. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django
4. https://www.fullstackpython.com/django.html
5. https://codelabs.developers.google.com/

❖ **Software:**

1. Python IDLE
2. Anaconda Python
3. PyCharm

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 7)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE441: BIG DATA ANALYTICS

Credits and Hours:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/week	3	4	0	7	**5**
Marks	100	100	0	200	

Pre-requisite courses:

- Linux Operating System
- Database Management System

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Big Data and Analytics	02
2.	Data Collection, Sampling and Preprocessing	06
3.	Predictive Analytics, Descriptive Analytics, Survival Analysis	08
4.	Introduction to Hadoop and Hadoop Architecture	08
5.	HDFS, HIVE and HIVEQL, HBASE	08
6.	Apache Spark and MongoDB	08
7.	Big Data Applications and Visualization	05
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Big Data and Analytics**	**02 Hours**	**4%**
	Introduction to Big Data, Big Data Characteristics, Types of Big Data, Traditional Versus Big Data Approach, Technologies Available for Big Data, Infrastructure for Big Data, Use of Data Analytics, Big Data Challenges.		
2.	**Data Collection, Sampling and Preprocessing**	**06 Hours**	**13%**

	Types of Data Sources Sampling, Types of Data Elements ,Visual Data Exploration and Exploratory Statistical Analysis, Missing Values, Outlier Detection and Treatment, Standardizing Data, Categorization, Weights of Evidence Coding, Variable Selection, Segmentation		
3.	**Predictive Analytics, Descriptive Analytics & Survival Analysis**	**08 Hours**	**18%**
	Predictive Analytics: Target Definition, Linear Regression, Logistic Regression, Decision Trees, Neural Networks, Support Vector Machines, Ensemble Methods, Multiclass Classification Techniques, Evaluating Predictive Models Descriptive Analytics: Association Rules, Sequence Rules, Segmentation Survival Analysis: Survival Analysis Measurements, Kaplan Meier Analysis, Parametric Survival Analysis, Proportional Hazards Regression, Extensions of Survival Analysis Models, Evaluating Survival Analysis Models		
4.	**Introduction to Hadoop and Hadoop Architecture**	**08 Hours**	**18%**
	Big Data – Apache Hadoop & Hadoop EcoSystem, Moving Data in and out of Hadoop – Understanding inputs and outputs of MapReduce -, Data Serialization		
5.	**HDFS, HIVE AND HIVEQL, HBASE**	**08 Hours**	**18%**
	HDFS-Overview, Installation and Shell, Java API; Hive Architecture and Installation, Comparison with Traditional Database, HiveQL Querying Data, Sorting And Aggregating, Map Reduce Scripts, Joins & Sub queries, HBase concepts, Advanced Usage, Schema Design, Advance Indexing, PIG, Zookeeper , how it helps in monitoring a cluster, HBase uses Zookeeper and how to Build Applications with Zookeeper		
6.	**Apache Spark, MongoDB and Neo4j**	**08 Hours**	**18%**
	Introduction to Data Analysis with Spark, Downloading Spark and Getting Started, Programming with RDD, Spark SQL, Spark Streaming. Introduction to MongoDB key features, Core Server tools, MongoDB		

	through the JavaScript's Shell, Creating and Querying through Indexes, Document-Oriented, principles of schema design, Constructing queries on Databases, collections and Documents , MongoDB Query Language		
7.	**Graph Analytics and Data Visualization**	**05 Hours**	**11%**
	Apache Spark GraphX: Property Graph, Graph Operator, SubGraph, Triplet, Neo4j: Modeling data with Neo4j, Cypher Query Language: General clauses, Read and Write clauses. Big Data Visualization with D3.js, Kibana and Grafana		

Course Outcome (COs):

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

CO1	Understand the key issues in big data management and its associated applications in intelligent business and scientific computing
CO2	Acquire fundamental enabling techniques and scalable algorithms like Hadoop, Map Reduce and NO SQL in big data analytics.
CO3	Interpret business models and scientific computing paradigms and apply software tools for big data analytics.
CO4	Achieve adequate perspectives of big data analytics in various applications like recommender systems and social media applications.
CO5	Evaluate and apply appropriate principles, techniques and theories to large-scale data science problems using various databases with analytics and visualizations.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
    1. Bart Baesens , Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, ,Wiley, 2014
- **Reference book:**
    1. Xyz Dirk Deroos et al., Hadoop for Dummies, Dreamtech Press, 2014.
    2. Chuck Lam, Hadoop in Action, December, 2010.
    3. Leskovec, Rajaraman, Ullman, Mining of Massive Datasets, Cambridge University Press.
    4. I.H. Witten and E. Frank, Data Mining: Practical Machine learning tools and techniques.
- **Web material:**
    1. https://cognitiveclass.ai/
    2. https://codelabs.developers.google.com/
- **Software & Platform:**
    1. R & SPSS
    2. Hadoop, HBase, Hive, Pig, Spark
    3. Casandra, Neo4j, NoSQL

CE442: DESIGN OF LANGUAGE PROCESSOR

Credits and Hours:

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

Pre-requisite courses:

- Digital Electronics
- Operating System
- Theory of Computation

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Overview of Language Processors & Translators	10
2.	Introduction To Compilers	20
3.	Symbol-Table Management	05
4.	Static & Dynamic Memory Allocation & Memory management	06
5.	Semantic Analysis & Intermediate Code Generation	05
6.	Code Optimization	07
7.	Code Generation	07
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	90

Detailed Syllabus:

1.	**Overview of Language Processors & Translators**	**12 Hours**	**20%**
	Language processing activities, fundamental of language processing, Operating System, Interpreter vs compiler Pre-processor & Macro Processors – Subroutine, Macro definition and call, Macro expression, nested macro call, Advanced macro facilities, design of macro pre-processor Loaders - Compile & go, Absolute, Bootstrap, Relocating, Linking loader Linkers - Relocation of Linking Concept, Design of Linker,		

	Linker for MS DOS, Linking for overlays, Linkage editor Assemblers - Elements of Assembly Language Programming, Assembly Scheme, Single pass Assembler, Two pass assembler, Data structure of Assemblers		
2.	**Introduction To Compilers**	**20 Hours**	**35%**
	Pass and Phases of Compiler, grouping of phases, Compiler Contraction tools Lexical analyser - Roles of lexical analyser, input buffering, tokens, Regular Expression, finite Automata Syntax analyser - Context free grammar, Ambiguous grammar, Top-down parsing, Bottom-up parsing, LEX, YACC		
3.	**Symbol-Table Management**	**02 Hours**	**05%**
	Data structures to implement symbol table, Symbol Attributes, Symbol-Table entries, Local Symbol Table management, Global Symbol Table Structure, Storage bindings and Symbolic Registers		
4.	**Static & Dynamic Memory Allocation & Memory management**	**06 Hours**	**09%**
	Data descriptors - Static and Dynamic storage allocation – Storage allocation and access in block structured programming languages – Array allocation and access- Compilation of expressions – Handling operator priorities – Intermediate code forms for expressions – code generator., Register Usage, The run-time stack, Parameter passing disciplines, Code sharing and Position-Independent code		
5.	**Semantic Analysis & Intermediate Code Generation**	**06 Hours**	**09%**
	Intermediate Languages, Declarations, Assignment Statements, Boolean Expressions, Case Statements, Back patching, Procedure Calls		
6.	**Code Optimization**	**07 Hours**	**11%**
	The Principal Sources of Optimization, Optimization of Basic Blocks, Loops in Flow Graphs, Introduction to Global Data-Flow Analysis, Iterative Solution of Data-Flow Equations,		

	Code-Improving Transformations, Dealing with Aliases, Data-Flow Analysis of Structured Flow Graphs, Efficient Data-Flow Algorithms, A Tool for Data-Flow Analysis, Estimation of Types, Symbolic Debugging of Optimized Code		
7.	**Code Generation**	**07 Hours**	**11%**
	Issues in the Design of a Code Generator, The Target Machine, Run-Time Storage Management, Basic Blocks and Flow Graphs, Next-Use Information, A Simple Code Generator, Register Allocation and Assignment, The DAG Representation of Basic Blocks, Peephole Optimization, Generating Code from DAGs, Dynamic Programming Code-Generation Algorithm, Code-Generator Generators		

Course Outcome (COs):

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

CO1	Understand design and processing of different language processor, loaders and linkers
CO2	Design top-down and bottom-up parsers
CO3	Identify different memory management schemes of language processors
CO4	Develop semantic analysis scheme to generate intermediate code
CO5	Apply different code optimization techniques
CO6	Develop algorithms to generate code for a target machine

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Alfred Aho, Ravi Sethi, Jeffrey D Ullman, "Compilers Principles, Techniques and- Tools", Pearson Education Asia.
  2. M. Dhamdhere, "System Programming and Operating Systems", Tata McGraw-Hill.
  3. Steven S. Muchnick. Advanced Compiler Design and Implementation
- **Reference book:**
  1. Allen I. Holub "Compiler Design in C", Prentice Hall of India.
  2. C. N. Fischer and R. J. LeBlanc, "Crafting a compiler with C", Benjamin Cummings.
  3. J.P. Bennet, "Introduction to Compiler Techniques", Second Edition, Tata McGraw-Hill
  4. HenkAlblas and Albert Nymeyer, "Practice and Principles of Compiler Building with C", PHI.
  5. Kenneth C. Louden, "Compiler Construction: Principles and Practice", Thompson Learning.
  6. Compiler Construction by Kenneth. C. Louden, Vikas Pub
- **Web material:**
  1. http://compilers.iecc.com/crenshaw
  2. http://www.compilerconnection.com
  3. http://dinosaur.compilertools.net
  4. http://pltplp.net/lex-yacc
- **Software:**
  1. LEX
  2. YACC

CE443: Cloud Computing

Credits and Hours:

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

Pre-requisite courses:

- Operating System
- Networking

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Cluster Computing, Grid Computing Systems and Resource Management at Glance	08
2.	Fundamental of Virtualization	06
3.	Fundamental Concepts and Models	06
4.	Cloud-Enabling Technology	05
5.	Fundamental Cloud Architectures	07
6.	Advanced Cloud Architectures	08
7.	Implementation of Cloud	05
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Cluster Computing, Grid Computing Systems and Resource Management at Glance**	**08 Hours**	**18%**
	Introduction, Eras of Computing, Scalable Parallel Computer Architectures, Towards Low Cost Parallel Computing and Motivations , A Cluster Computer and its Architecture, Clusters Classification, Commodity Components for Clusters, Grid Architecture and Service Modelling, Grid Projects and Grid Systems Built, Grid Resource Management and Brokering, Software and Middleware for Grid Computing,		

	Grid Application Trends		
2.	**Fundamental of Virtualization**	**06 Hours**	**12%**
	Type of Virtualization, Virtualization Technologies, Virtualizes your Environment, Managing Virtualization Environment, Storage Virtualization, Dockers		
3.	**Fundamental Concepts and Models**	**06 Hours**	**15%**
	Roles and Boundaries, Cloud Characteristics, Cloud Delivery Models, Cloud Deployment Models		
4.	**Cloud-Enabling Technology**	**05 Hours**	**14%**
	Broadband Networks and Internet Architecture, Data centre Technology, Virtualization Technology, Web Technology, Multitenant Technology, Service Technology		
5.	**Fundamental Cloud Architectures**	**07 Hours**	**16%**
	Workload Distribution Architecture, Resource Pooling Architecture ,Dynamic Scalability Architecture, Elastic Resource Capacity Architecture, Service Load Balancing Architecture, Cloud Bursting Architecture, Elastic Disk Provisioning Architecture, Redundant Storage Architecture		
6.	**Advanced Cloud Architectures**	**08 Hours**	**17%**
	Hypervisor Clustering Architecture ,Load Balanced Virtual Server Instances Architecture, Non-Disruptive Service Relocation Architecture, Zero Downtime Architecture ,Cloud Balancing Architecture ,Resource Reservation Architecture, Dynamic Failure Detection and Recovery Architecture		
7.	**Implementation of Cloud**	**05 Hours**	**08%**
	Study of Cloud computing Systems like Amazon EC2 and S3, Google App Engine, and Microsoft Azure, Build Private/Hybrid Cloud using open source tools, Deployment of Web Services from Inside and Outside a Cloud Architecture. MapReduce and its extensions to Cloud Computing, HDFS, and GFS		

Course Outcome (COs):

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

CO1	Assess and examine advantages and disadvantages of virtualization technology.
CO2	Compose services in a distributed computing environment to achieve tasks relevant to a knowledge-based business or public service
CO3	Evaluate a set of business requirements to determine suitability for a cloud computing delivery model.
CO4	Explore the various cloud computing architectures and paradigms.
CO5	Deployment of cloud and identify security implications in cloud computing.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	2	2	1	0	1	-	-	-	1	2	-
CO2	3	2	3	1	3	2	1	-	-	-	1	1	2	1
CO3	3	2	2	3	3	1	2	-	1	1	-	-	2	1
CO4	3	1	1	2	2	-	-	2	1	-	-	-	1	2
CO5	3	1	3	1	3	2	1	1	-	1	-	1	1	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 "-"

Recommended Study Material:

- Text Books:
  1. Thomas Erl, Zaigham Mahmood,and Ricardo Puttini, "Cloud Computing Concepts, Technology & Architecture", Prentice Hall
  2. Kai Hwang, Geoffrey C.," Distributed and Cloud Computing", Morgan Kaufmann is an imprint of Elsevier
  3. Navin Sabharwal, Ravi Shankar "Apache CloudStack Cloud Computing" PACKT Publishing

❖ **Reference Books**:

1. Ravi Shankar, Navin Sabharwa "Cloud Computing First Steps: Cloud Computing for Beginners" Create Space Independent Publishing Platform
2. Rajkumar Buyya, James Broberg, Andrzej Goscinski "Cloud Computing: Principles and Paradigms" Wiley
3. Judith Hurwitz, Robin Bloor "Cloud Computing For Dummies" , for Dummies

❖ **Web material:**

1. http://www.console.cloud.google.com
2. http://www.qwicklabscom
3. http://codelabs.developers.google.com
4. http://www.docker.com

❖ **Software/Platform:**

1. NetBeans
2. Eclipse
3. .NET
4. GCP
5. Amazone
6. Microsoft

CE444: INTERNET OF THINGS

Credits and Hours:

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

Pre-requisite courses:

- Networking
- Embedded System

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Introduction and evolution of IoT	04
2.	Organisation and primary components of IoT systems	08
3.	A reference IoT architecture	10
4.	Design issues for the IoT edge	12
5.	Security, trust, and privacy issues in IoT	08
6.	IoT case studies	03
	Total hours (Theory) :	45
	Total hours (Lab) :	30
	Total hours :	75

Detailed Syllabus:

1.	**Introduction and evolution of IoT**	**04 Hours**	**09 %**
	Introduction, Internet of Things Definition Evolution		
2.	**Organisation and primary components of IoT systems**	**08 Hours**	**18 %**
	Structure of IoT systems, IoT backend modules, IoT gateways, The IoT edge		
3.	**A reference IoT architecture**	**10 Hours**	**22%**
	Design principles and design requirements for the reference architecture, Real-world constraints		

4.	**Design issues for the IoT edge**	**12 Hours**	**27%**
	Sensors and actuators for IoT systems, Interoperability and reliability issues Communication protocols and protocol stacks for the edge devices, Hardware security for edge devices		
5.	**Security, trust, and privacy issues in IoT**	**08 Hours**	**18%**
	Identity management of IoT edge devices		
6.	**IoT case studies**	**03 Hours**	**7%**
	Smart grid, Home automation, Industrial IoT		

Course Outcome (COs):

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

CO1	Interpret the vision of IoT from current business needs.
CO2	Compare and contrast the use of devices, gateways and data management in IoT.
CO3	Demonstrate a critical understanding of key technologies in sensing, data transmission and data processing for IoT.
CO4	Examine of the technologies and the standards relating to the Internet of Things and analyze basic protocols in sensor network
CO5	Evaluation of privacy management, challenges in designing and securing an IoT system.
CO6	Illustrate the application of IoT in industrial automation and identify real world design Constraints.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Internet of Things: principles and paradigms, Buyya, Rajkumar and Amir Vahid Dasterdji (eds.), Morgan Kaufmann, 2016.
  2. From Machine-to-Machine to the Internet of Things: introduction to a new age of intelligence, Holler, Jan et al, Academic Press, 2014.
- **Reference book:**
  1. Building Internet of Things with the Arduino, Doukas, Charalampos, Create Space Independent Publishing Platform, 2012.
  2. Francis daCosta, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", 1st Edition, Apress Publications, 2013.
- **Web material:**
  1. http://web.mit.edu/professional/digital-programs/courses/IoT/phone/index.html
  2. https://swayam.gov.in/nd1_noc19_cs65/preview
  3. https://www.edureka.co/blog/iot-tutorial/
  4. http://www.steves-internet-guide.com/internet-of-things/
- **Software:**
  1. Contiki OS
  2. Node-Red
  3. Arduino

CE471: BLOCKCHAIN TECHNOLOGY (ELECTIVE- III)

Credits and Hours:

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

Pre-requisite courses:

- Basic Information Security

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Fundamentals Behind Blockchain	07
2.	Blockchain Overview	06
3.	Block-Chain Consensus Mechanisms	12
4.	Introduction to BitCoin	12
5.	Solidity Essential	08
6.	Decentralized Application	10
7.	Blockchain Research issues, challenges and use cases	05
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	90

Detailed Syllabus:

1.	**Fundamentals Behind Blockchain**	**07 Hours**	**11%**
	Symmetric key cryptography, Public key cryptography, Digital Signature, Cryptographically Secured Hash Functions, Cryptographically Secured Chain of Blocks, Merkle Trees.		
2.	**Blockchain Overview**	**06 Hours**	**10%**
	Evaluation of Blockchain Technology, Distributed Systems, The History of Blockchain and Bitcoin, Types of Block-		

	Chain		
3.	**Block-Chain Consensus Mechanisms**	**12 Hours**	**20%**
	Practical Byzantine fault tolerance algorithm, Proof of Work, Proof of Stake, Proof of Authority, Proof of Elapsed time		
4.	**Introduction to BitCoin**	**12 Hours**	**20%**
	Digital keys and addresses, Transactions, Mining, The Bitcoin network, Wallets, Bitcoin payments		
5.	**Solidity Essential**	**08 Hours**	**13%**
	Using the Remix IDE, Data Types and Functions, Creating Inline Assembly Functions, Mappings, Modifiers, Structs, and More, Generating ERC-20 Tokens, Extending Token Security (ERC-223),Deploying the ERC-20 Token Contract.		
6.	**Decentralized Application**	**10 Hours**	**17%**
	DAPP using Ehtereum, DAPP using Hyperledger.		
7.	**Blockchain Research issues, challenges and use cases**	**05 Hours**	**09%**
	KYC case study, Land Registry, Supply chain.		

Course Outcome (COs):

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

CO1	Understand the Basic Cryptography behind the Blockchain Technology and Bitcoin.
CO2	Define the structure of a Blockchain and classify why and when it is better than a simple distributed database.
CO3	Analyse the consensus mechanisms in a Blockchain Technology and critically assess its applicability in Blockchain based application.
CO4	Analyse to what extent smart and self-executing contracts can benefit automation, governance and transparent environment.
CO5	Design decentralized distributed application and measure the performance of Blockchain against centralized system.
CO6	Attain awareness of the new challenges that exist in monetizing businesses around Blockchain.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

- **Text book:**
  1. Imran Bashir , Mastering Blockchain ,Packt Second Edition, 2018.
  2. Andreas M. Antonopoulos, Mastering Bitcoin, O'Reilly, Second Edition
- **Reference book:**
  1. Samanyu, Blockchain Developer's Guide, Packt, 2018.
  2. Mark Watney, Blockchain for Beginners: The Complete Step by Step Guide to Understanding Blockchain Technology, July, 2017.
  3. Don Tapscott, Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World, Hardcover, May 2016.
  4. Xun (Brian) Wu, Hyperledger Cookbook, Packt.
  5. Mayukh Mukhopadhyay, Ethereum Smart Contract Development, Packt,2018.
- **Web material:**
  1. https://medium.com/topic/blockchain
- **Software:**
  1. Ethereum
  2. Hyperledger Fabric 1.4
  3. Truffle
  4. Solidity

CE472: WIRELESS COMMUNICATION & MOBILE COMPUTING

(ELECTIVE- III)

Credits and Hours:

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

Pre-requisite courses:

- Basic Knowledge of Computer Networks

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Wireless Communication Fundamentals	03
2.	Telecommunication Systems	13
3.	Wireless LAN	18
4.	Mobile Network Layer	13
5.	Transport and Application Layer	10
6.	Data Dissemination and Management	03
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	90

Detailed Syllabus:

1.	**Wireless Communication Fundamentals**	**03 Hours**	**5%**
	Introduction, Wireless transmission – Frequencies for radio transmission – Signals – Signal Propagation ,Multiplexing – Modulations – Spread spectrum,MAC – SDMA – FDMA – TDMA – CDMA – Cellular Wireless Networks		
2.	**Telecommunication Systems**	**13 Hours**	**25%**
	GSM: Mobile services, System architecture, Radio interface, Protocols, Localization And Calling, Handover, Security, New data services; DECT: System architecture, Protocol architecture; TETRA, UMTS and IMT-2000		
3.	**Wireless LAN**	**18 Hours**	**30%**
	Wireless LAN – IEEE 802.11 - Architecture – services – MAC – Physical layer,IEEE 802.11a - 802.11b – 802.11n		

	standards,Bluetooth,Hyperlan, Wi-Fi, WiMax - Overview		
4.	**Mobile Network Layer**	**13 Hours**	**20%**
	Mobile IP,Dynamic Host Configuration Protocol, Routing Protocols – DSDV – DSR – Alternative Metrics		
5.	**Transport and Application Layer**	**10 Hours**	**15%**
	Traditional TCP,Classical TCP improvements – WAP, WAP 2.0		
6.	**Data Dissemination and Management**	**03 Hours**	**05%**
	Challenges, Data dissemination, Mobile data replication, Mobile data caching, Mobile cache maintenance, mobile web caching, caching in ad hoc networks.		

Course Outcome (COs):

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

CO1	Classify the fundamental concepts of Wireless Networks and its access techniques.
CO2	Understand and identify the GSM, GPRS and Bluetooth software model for mobile computing.
CO3	The ability to develop applications that are mobile-device specific and demonstrate current practice in mobile computing contexts.
CO4	Understanding of the characteristics and limitations of mobile hardware devices including their user-interface modalities.
CO5	Able to promote the awareness of the life-long learning, business ethics, professional ethics and current marketing scenarios in wireless network.

Course Articulation Matrix:

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

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 "-"

Recommended Study Material:

❖ **Text Books:**

1. "Mobile Computing: Technology, Applications and Service Creation" by Asoke K Talukder and Roopa R Yavagal, TMH,ISBM: 0-07-058807-4

❖ **Reference Materials:**

1. Jochen Schiller, "Mobile Communications", PHI/Pearson Education, Second Edition, 2003.
2. William Stallings, "Wireless Communications and Networks", PHI/Pearson Education, 2002.
3. Kaveh Pahlavan, Prasanth Krishnamoorthy, "Principles of Wireless Networks", PHI/Pearson Education, 2003.
4. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, "Principles of Mobile Computing", Springer, New York, 2003.
5. Hazysztof Wesolowshi, "Mobile Communication Systems", John Wiley and Sons Ltd, 2002
6. Research papers from IEEE, Springer etc.

❖ **Web Materials:**

1. www.ietf.org – For drafts
2. www.ieee.org – For standards and technical research papers

❖ **Software:**

1. OMNET++
2. SUMO
3. MATLAB
4. NS3

CE473 MACHINE LEARNING (ELECTIVE- III)

Credits and Hours:

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

Pre-requisite courses:

- Probability, linear algebra, calculus and programming language

Outline of the Course:

Sr. No.	Title of the unit	Minimum number of hours
1.	Fundamental concepts and Statistical Learning Techniques	10
2.	Neural Networks	08
3.	Bayesian Learning	06
4.	Supervised and Unsupervised Learning	15
5.	Reinforcement learning	04
6.	Kernel Methods	05
7.	Deep Neural Networks	12
	Total hours (Theory) :	60
	Total hours (Lab) :	30
	Total hours :	90

Detailed Syllabus:

1.	**Fundamental concepts and Statistical Learning Techniques**	**10 hours**	**17%**
	Introduction to Data science, Theory and practices in machine learning, Designing a Learning System, Issues in Machine Learning, Applications of ML, Global Developments of ML, Key challenges to adoption of ML in India. Statistical Learning Techniques: Descriptive statistics, Simple Linear Regression, ANOVA, Logistic Regression, Multi Linear regression, Correlation, Moving Average, Random Number Generation, Histogram		

	Smoothing, Sampling, Regularization, Rank Percentile.		
2.	**Neural Networks**	**08 Hours**	**14%**
	Neurons and biological motivation. Linear threshold units. Perceptrons: representational limitation and gradient descent training, Perceptron learning rule, Hebbian learning rule, Delta Learning rule, Multilayer networks and Backpropagation Learning Algorithm, Feed Forward, Activation Functioning, Types of Neural Network Architecture.		
3.	**Bayesian Learning**	**06 hours**	**10%**
	Bayes Theorem, Bayes Theorem and Concept Learning, Maximum Likelihood and Least Squared Error Hypothesis, Maximum likelihood hypothesis for Predicting probabilities, Minimum Description Length Principle, Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier, Bayesian Belief Network, EM Algorithm.		
4.	**Supervised and Unsupervised Learning**	**15 hours**	**25%**
	Supervised Learning: Classification, Decision Tree, Naïve Bias, Support Vector Machine, Neural Network, K- Nearest Neighbour. Unsupervised Learning: Clustering, Nonparametric Methods, K-means, Hierarchical clustering, Density based clustering		
5.	**Reinforcement learning**	**04 hours**	**06%**
	Q Learning, Non deterministic rewards and Actions.		
6.	**Kernel Methods**	**05 hours**	**8%**
	Support Vector Machine, Sparse kernel machines, Bias-Variance trade-off. Regularization and model/feature selection, Sampling Methods.		
7.	**Deep Neural Networks**	**12 hours**	**20%**
	Introduction to Deep Learning, Deep Neural Network, Restricted Boltzmann machine, Convolution Neural Network, AutoEncoders, Deep Belief Network, Recurrent Neural		

	Network, Transfer learning.		

Course Outcome (COs):

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

CO1	Apply basic concepts of Machine Learning and Understanding of standard learning algorithms.
CO2	Analyse mathematical modelling of various Machine Learning algorithms.
CO3	Understanding challenges of machine learning like data characteristics, model selection, and model complexity.
CO4	Identify strengths and weaknesses of machine learning techniques suitable for a given problem domain and data set.
CO5	Design and implement of various machine learning algorithms in a range of real-world applications.
CO6	Evaluate and interpret the results of learning algorithms.

Course Articulation Matrix:

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

Recommended Study Material:

- **Text book:**
  1. Xyz Machine Learning, Tom Mitchell, McGraw Hill, 1997. ISBN 0070428077
  2. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, 2004

❖ **Reference book:**

1. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.
2. Richard O. Duda, Peter E. Hart & David G. Stork, "Pattern Classification. Second Edition", Wiley & Sons, 2001.
3. Trevor Hastie, Robert Tibshirani and Jerome Friedman, "The elements of statistical learning", Springer, 2001.
4. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", MIT Press, 1998.

❖ **Web material:**

1. https://www.youtube.com/watch?v=f-oHSmB48rY&list=PLKvX2d3IUq586Ic9gIhZj6ubpWV-OJfl4
2. https://www.youtube.com/watch?v=CS4cs9xVecg&list=PLkDaE6sCZn6Ec-XTbcX1uRg2_u4xOEky0
3. https://www.youtube.com/watch?v=UzxYlbK2c7E
4. https://www.youtube.com/playlist?list=PLAwxTw4SYaPkQXg8TkVdIvYv4HfLG7SiH

❖ **Software:**

1. Scikit Learn
2. PyTorch
3. TensorFlow
4. Colab
5. Keras

CE448: SOFTWARE GROUP PROJECT-V

Credits and Hours:

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

Pre-requisite courses:

- Programming Knowledge

Outline of the Course:

- Student at the beginning of a semester may be advised by his/her supervisor (s) for recommended courses.
- Students will work together in a team (at most three) with any programming language.
- 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 of continuous 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.

  m. Project Synopsis

  n. Software Requirement Specification

  o. SPMP

  p. Final Project Report

  q. Project Setup file with Source code

  r. Project Presentation (PPT)

- A student has to produce some useful outcome by conducting experiments or project work.

Total hours (Theory) : 00

Total hours (Lab) : 60

Total hours : 60

Course Outcome (COs):

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

CO1	Identify and define the computing requirements of a problem to propose its appropriate solution.
CO2	Correlate knowledge of different subjects and apply it to implement solution of the problem.
CO3	Apply engineering and management principles to achieve project goal.
CO4	Prepare technical report and deliver presentation by applying different visualization tools and evaluation metrics.
CO5	Able to communicate effectively with a range of audiences.
CO6	Recognition of the need for and an ability to engage in continuing professional development.
CO7	Able to work and coordinate with different kind of people in the team

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	3	-	2	-	-	-	-	2	-	-	-	3	-
CO2	3	-	1	1	3	1	1	-	-	-	-	2	1	1
CO3	-	-	3	-	3	1	1	-	2	-	3	2	3	-
CO4	-	-	-	-	3	-	2	-	2	3	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	3	-	-	-	-
CO6	-	-	-	-	-	-	3	-	-	-	-	2	-	2
CO7	-	-	-	-	-	-	-	2	3	3	-	2	-	2

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 "-"

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines & Journals of related topics
- **Web material:**
    1. www.ieeexplore.ieee.org
    2. www.sciencedirect.com
    3. www.elsevier.com
    4. http://spie.org/x576.xml
- **Software:**
    1. ASP.NET
    2. PYTHON/MATLAB
    3. PHP
    4. ANDROID/IOS

CE446: SUMMER INTERNSHIP-II

Credits and Hours:

Teaching Scheme	Project	Practical	Tutorial	Total	Credit
Hours	90	-	-	90	**3**
Marks	**150**	-	-	**150**	

Objectives of the Course:

- To get familiar with modern tools and technologies use in company/industry/organization
- To get involved in design, development and testing practices followed in the company/industry/organization
- To enhance their soft-skills, presentation skills, interpersonal skills, documentation skills and office etiquettes required to sustain in company/industry/organization environment
- To participate in teamwork and preferably as part of a multi-disciplinary team
- To make them aware about company/industry/organization best practices, processes and regulations.
- To make them more productive, consistent and punctual.

Outline of the Course:

1.	**Instructional Method and Pedagogy**
	• Summer internship shall be at least 90 hours during the summer vacation only. • Department/Institute will help students to find an appropriate company/industry/organization for the summer internship. • The student must fill up and get approved a Summer Internship Acceptance form by the company and provide it to the Coordinator of the department within the specified deadline. • Students shall commence the internship after the approval of the department Coordinator. Summer internships in research centers is also allowed. • During the entire period of internship, the student shall obey the rules and regulations of the company/industry/organization and those of the University. • Due to inevitable reasons, if the student will not able to attend the internship

	for few days with the permission of the supervisor, the department Coordinator should be informed via e-mail and these days should be compensated later.
	• The student shall submit two documents to the Coordinator for the evaluation of the summer internship:
	  • Summer Internship Report
	  • Summer Internship Assessment Form
	• Upon the completion of summer internship, a hard copy of "Summer Internship Report" must be submitted through the presentation to the Coordinator by the first day of the new term.
	• The report must outline the experience and observations gained through practical internship, in accordance with the required content and the format described in this guideline. Each report will be evaluated by a faculty member of the department on a satisfactory/unsatisfactory basis at the beginning of the semester.
	• If the evaluation of the report is unsatisfactory, it shall be returned to the student for revision and/or rewriting. If the revised report is still unsatisfactory the student shall be requested to repeat the summer internship.

2.	**Format of Summer Internship Report**
	The report shall comply with the summer internship program principles. Main headings are to be centered and written in capital boldface letters. Sub-titles shall be written in small letters and boldface. The typeface shall be Times New Roman font with 12pt. All the margins shall be 2.5cm. The report shall be submitted in printed form and filed. An electronic copy of the report shall be recorded in a CD and enclosed in the report. Each report shall be bound in a simple wire vinyl file and contain the following sections: • Cover Page • Page of Approval and Grading • Abstract page: An abstract gives the essence of the report (usually less than one page). Abstract is written after the report is completed. It must contain the purpose and scope of internship, the actual work done in the plant, and

conclusions arrived at.

- TABLE OF CONTENTS (with the corresponding page numbers)
- LIST OF FIGURES AND TABLES (with the corresponding page numbers)
- DESCRIPTION OF THE COMPANY/INDUSTRY/ORGANISATION: Summarize the work type, administrative structure, number of employees (how many engineers, under which division, etc.), etc. Provide information regarding
    - Location and spread of the company
    - Number of employees, engineers, technicians, administrators in the company
    - Divisions of the company
    - Your group and division
    - Administrative tree (if available)
    - Main functions of the company
    - Customer profile and market share
- INTRODUCTION: In this section, give the purpose of the summer internship, reasons for choosing the location and company, and general information regarding the nature of work you carried out.
- PROBLEM STATEMENT: What is the problem you are solving, and what are the reasons and causes of this problem.
- SOLUTION: In this section, describe what you did and what you observed during the summer internship. It is very important that majority of what you write should be based on what you did and observed that truly belongs to the company/industry/organization.
- CONCLUSIONS: In the last section, summarize the summer internship activities. Present your observations, contributions and intellectual benefits. If this is your second summer internship, compare the first and second summer internships and your preferences.
- REFERENCES: List any source you have used in the document including books, articles and web sites in a consistent format.
- APPENDICES: If you have supplementary material (not appropriate for the main body of the report), you can place them here. These could be schematics,

	algorithms, drawings, etc. If the document is a datasheet and it can be easily accessed from the internet, then you can refer to it with the appropriate internet link and document number. In this manner you don't have to print it and waste tons of paper.	
	Total hours (Project) :	**90**
	Total hours :	**90**

Course Outcome (COs):

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

CO1	Ability to integrate existing and new technical knowledge for industrial application.
CO2	Executing work with team and teammates from other disciplines
CO3	Get practices and experience related to professional and ethical issues in the work environment
CO4	Experience of demonstrating the impact of the internship on their learning and professional development.
CO5	Understanding of lifelong learning processes through critical reflection of internship experiences.

Course Articulation Matrix:

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

Recommended Study Material:

- **Reference book:**
    1. Books, Magazines & Journals of related topics
- **Web material:**
    1. www.ieeexplore.ieee.org
    2. www.sciencedirect.com
    3. www.elsevier.com
    4. http://spie.org/x576.xml
- **Software**
    1. ASP.NET
    2. PYTHON/MATLAB
    3. PHP
    4. ANDROID/IOS
    5. FLUTTER
    6. NODE/REACT NATIVE

B. Tech. (Computer Engineering) Programme

SYLLABI

(Semester – 8)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CE447: SOFTWARE PROJECT MAJOR

Credits and Hours:

Teaching Scheme	Theory	Practical	Total	Credit
Hours/week	0	36	36	**20**
Marks	0	600(250+350)	600	

Pre-requisite courses:

- Software Engineering and other relevant courses, Development tools and languages, soft skill.

Outline of the Course:

- Students work with industry/organization4-to-6 months for development or research project.
- The definitions are verified and guidance are given to take the project at next level and can help industry, society or environment.
- The external guide (at industry/Organization) and internal guide (Institute faculty) continuously monitor the students' work and project file is maintained per group to document all the measurable work.
- Project work should include whole SDLC/Agile of development of software / hardware system as solution of particular problem by applying principles of Software Engineering.
- Project is evaluated twice during the semester by internal faculty of the university as a part of continuous evaluation.
- Final evaluation at the end of the semester is done by Industry experts or faculties from reputed university.
- Students have to submit SRS, SPMP, Design documents, Code and Test Cases in form of Project report.
- The feedback of the students' by external guide and Evaluator are taken during the semester to improve the teaching learning and evaluation process.

Course Outcome (COs):

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

CO1	Identify and justify/analyse the requirements of the projects with enhancement of the required tools and technology by individual and team.
CO2	Solve challenging projects for commercial, societal and environment benefit.
CO3	Apply the engineering knowledge to full fill the requirements of the projects pertaining to any discipline.
CO4	Explain the importance of planning, documentation, punctuality and work ethics.
CO5	Document the work which is carried out in proper format with industry standards.
CO6	Showcase the soft skill.

Course Articulation Matrix:

	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	2	3	-	1	-	3	1	1	1	3	1
CO2	2	-	3	-	-	3	2	-	-	-	-	-	2	2
CO3	3	2	1	1	1	-	-	-	1	-	2	1	3	1
CO4	3	1	2	-	-	-	-	3	-	-	1	2	2	3
CO5	3	-	2	2	-	-	-	-	-	2	-	3	1	1
CO6	1	-	1	1	-	-	-	3	1	3	1	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 "-"

Recommended Study Material (Suggested by Internal or External guide):

- Reading Materials, web materials, Project reports with full citations.
- Books, magazines & Journals of related topics.
- Various software tools and programming languages compiler related to topic.

❖ **Web Materials**:

1. www.ieeexplore.ieee.org
2. www.sciencedirect.com
3. www.elsevier.com