ACADEMIC REGULATIONS, PROGRAM STRUCTURE AND SYLLABUS

COMPUTER SCIENCE AND ENGINEERING

For

B.TECH. FOUR YEARS DEGREE PROGRAM
(Applicable to the batches admitted from 2020-21)
(I To IV Semesters)

ADITYA ENGINEERING COLLEGE

(An Autonomous Institution)
Approved by AICTE, Affiliated to JNTUK & Accredited by NBA, NAAC with 'A' Grade
Recognized by UGC under the sections 2(f) and 12(B) of UGC act 1956
Aditya Nagar, ADB Road, SURAMPALEM - 533 437

ABOUT ADITYA ENGINEERING COLLEGE

ADITYA ENGINEERING COLLEGE (AEC) was established in 2001 at Surampalem, Kakinada, Andhra Pradesh in 180 Acres of pollution free and lush green landscaped surroundings by the visionaries of Aditya Academy who have been in the field of education since last 3 ½ decades, extending their relentless and glorious services.

AEC believes in the holistic development of society at large and is striving hard by putting its efforts in multi-disciplinary activities. The College shoulders the responsibility of shaping the Intellect, Character and Physique of every student, because it believes that these are rudimentary aspects for students to develop a humanized and harmonious society, and become meaningful architects of the nation as a whole.

Our vision is to impart quality education, in a congenial atmosphere, as comprehensive as possible, with the support of all the modern technologies and produce graduates and post graduates in engineering with the ability and passion to work wisely, creatively, and effectively for the welfare of the society. It is our endeavor to develop a system of Education which can harness students' capabilities, potentialities and the muscles of the mind thoroughly trained to enable it to manifest great feats of intellectualism.

AEC has International standards and aims to be a centre of excellence to produce principal architects of the future. AEC is in the field of education for enriching the knowledge of budding youth with innovative calibre and to equip them with competitive skills fit for job, fit for life.

<u>SALIENT FEATURES:</u>

- Autonomous status by UGC, Accredited by NBA & NAAC with A Grade.
- Rated as "SILVER" by AICTE for best industry linked institution under CII.
- Recognized by UGC under sections 2(f) &12(B).
- Recognized by JNTUK, Affiliating University as "Research Center".
- Rated Grade "A" by Govt. of AP.
- Students from 17 states across India & 15 foreign countries, 500+ International students.
- 66 Ph. Ds,50+Research Scholars,25Patents&600+ National/International Research Publications.

- An ISO 9001-2015 certified institution.
- Recognized by Scientific and Industrial Research Organizations (SIROs) Govt. of India.
- Incubation Centre Campus & only campus in the state to receive financial Assistance from Govt. of India.
- Infosys Campus Connect Institution.
- NASCOM certification training program Campus.
- Collaboration with Co-Cubes and AMCAT for Scientific Assessment.
- Nodal Center for Indian School of Business & AP Information Tech. Academy (APITA).
- Honoured with Best Placement Award by Chief Minister of AP.
- Only college in AP to receive Best Performance Award from Tech Mahindra for its outstanding achievement in campus placements.
- Special CRT Training from first year.
  - Received 13 Pratibha Awards from Govt. of AP at District Level.
- Got 1st prize in AICTE CHHATRA VISWAKARMA student research awards at all India level in Water and Irrigation Category and received award from Hon'ble Vice-President of India.
- Ranked 3rd amongst the cleanest Higher Educational Institutions in the Country by Ministry of Human Resources Development, Government India in the category Residential Colleges – AICTE.
- Awarded 2nd Rank and Certificate of Recognition under the AICTE's National Level UtkrishtSansthan Vishwakarma Award 2019 for its significant contributions in the growth and development of adopted village.
- MOU with Educational Consultants India Ltd., (EdCIL) and MOUs with 5 foreign universities.
- MOU with CL Educate to increase the research activities in students.
- On campus hostels for Boys and Girls with good infrastructure facilities.
- Transportation facility with 80+ buses.
- Best Rankings & Ratings
  - ➢ Ranked at Rank Band 26-50 in Atal Ranking of Institutions on Innovation Achievements (ARIIA).
  - ➢ Careers360-ranked AAA+ in India's Best Engineering Colleges in South Zone.
  - ➢ Times Engineers-10th Rank in AP. Dataquest ranked 78 in Top 100 T-Schools in India.
  - ➢ India Today-14th Rank in AP & 132nd Rank at All India Level.
  - ➢ THE WEEK-60th Rank in South Zone & 13th Rank in AP.
  - ➢ Ranked AAAA in Top Engineering Institute Ranking in India.
  - ➢ The Academic Insights-33rd Rank in top 50 Colleges in India.
  - ➢ Silicon India-4th rank in top 10 colleges in South India.
  - ➢ The Sunday Indian-One of the best 20 Engineering Colleges in India.
  - ➢ 4Ps-13 rank in India out of top 25 Engineering Colleges.
  - ➢ WCRC Leaders-Asia Top 100 colleges.
  - ➢ Higher Education Review-India's 34 rank.
  - ➢ Placed in Best 10 Agricultural Institutes in India.
- Established Applied Robo Controlled Lab (ARC) Siemens Centre of Excellence Campus, in association with APSSDC, Indo European Skilling Centers for Mechatronics and Industrial Robotics.

- Established 7 excellent engineering labs by incurring Rs. 12 crores for improving skill based training in the students funded by SIEMENS. The only campus to receive such huge amount from SIEMENS through APSSDC in the district.
- Recognized as CM Centre of Excellence Campus. APSSDC with the collaboration of Dassault Systems have established 4 prestigious labs under this program.
- Recognized as PMKVY-TI Centre by AICTE.
- Fellowship program with Stanford University, California, USA
- India’s first Microsoft Ed-vantage Platinum Campus, Microsoft Innovation Centre campus.
- Recognized by Computer Society of India (CSI) as Best Accredited Student Branch
- NPTEL Best Local Chapter Award with AA Grade and stood 39th position in India and 7th position in AP
- MOUs with CISCO Networking Academy, SAP, ORACLE Academy, Dell EMC, Red Hat Academy, VMWare IT Academy, Hacker Rank, Comp Tia, SAK Robotix Lab, ARM University, GIT Hub, Ui Path and more under Aditya’s Technical HUB (T-HUB)
- Campus with Student Start-ups & these Start-ups received appreciation from Chief Minister of AP.
- Received Gold medal from Governor of AP (through Red Cross) for conducting Blood Donation camps every year & highest No. of students donated blood in a single phase (2541 Students) in 2017-18
- Strong Network of ALUMNI
- 100% Wi-fi Campus, 100% Surveillance Campus under CC cameras and in campus Bank with 8 ATMs.
- Aditya was sanctioned an incubator named “Aditya Global Business Incubator (AGBI) from DST. AGBI was sanctioned two Coir Clusters as implementing Agency worth Rs. 7.5 Crore by MSME.
- AGBI was sanctioned Lively hood Business Incubator worth Rs. 1.1 Crore for establishing Food Processing Training Institute.

VISION & MISSION OF THE COLLEGE

VISION

To induce higher planes of learning by imparting technical education with International standards, Applied research, Creative ability and Value based instruction to emerge as a premier institute.

MISSION

Achieving academic excellence by providing globally acceptable technical education by forecasting technology through

- Innovative research &development
- Industry instituteinteraction
- Empoweredmanpower

VISION & MISSION OF THE DEPARTMENT

VISION

To produce competent professionals to become part of the industry and research organizations at the national and international levels through excellence in Computer Science & Engineering education and research.

MISSION

M1: Designing curriculum to meet the future challenges in Computer Science & Engineering and society by anticipating relevant trends.

M2: Inculcating the problem solving skills, leadership qualities in students and enable them to work in teams to become successful in their careers.

M3: Nurturing with Scientific Research in the field of Information Technology, enable students to involve in technological innovations.

M4: Transforming the Computer Science and Engineering department as a leader in imparting Computer Science and Engineering education and research by a committed faculty.

****

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

Graduates of the Program will

PEO 1	Successfully be employed in industry, government, or entrepreneurial endeavors and solve complex problems by the applications of Technologies to meet the needs of employers.
PEO 2	Adapt to new technologies, tools and methodologies, to assess and respond to the challenges of the changing environment and needs of the society by providing sustainable innovative solutions to upgrade the society forever.
PEO 3	Apply fundamental knowledge, making them fit to pursue higher education in leading University in India/abroad or computing as a career.
PEO 4	Demonstrate interpersonal skills, leadership ability and team building to achieve organization goals and serve society with professional ethics and integrity.

PROGRAM OUTCOMES (POs)

After successful completion of the program, the graduates will be able to

PO 1	Apply knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
PO 2	Identify, formulate, research literature and analyze complex engineering problems, reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
PO 3	Design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PO 4	Conduct investigations of complex problems using research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.
PO 5	Create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering activities, with an understanding of the limitations.
PO 6	Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.
PO 7	Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate knowledge of, and need for sustainable development.
PO 8	Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.
PO 9	Function effectively as an individual, and as a member or leader in diverse teams and in multidisciplinary settings.
PO 10	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.

PO 11	Demonstrate knowledge and understanding of engineering management principles and apply these to one's own work, as a member and leader in a team and to manage projects in multidisciplinary environments.
PO 12	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)

After successful completion of the program, the graduates will be able to

PSO 1	Apply modern computer languages, environments, and platforms leading to innovative career paths.
PSO 2	Design and develop software in Networking, Mobile, Cloud Computing, Security etc.,
PSO 3	Analyze the enormous data using data analytics tool and various programming languages.

Mission of the department – PEOs mapping

PEO's Statements		M1	M2	M3	M4
PEO 1:	Successfully be employed in industry, government, or entrepreneurial endeavors and solve complex problems by the applications of Technologies to meet the needs of employers.	3	3	3	3
PEO 2:	Adapt to new technologies, tools and methodologies, to assess and respond to the challenges of the changing environment and needs of the society by providing sustainable innovative solutions to upgrade the society forever.	3	3	3	3
PEO 3:	Apply fundamental knowledge, making them fit to pursue higher education in leading University in India/abroad or computing as a career.	3	3	3	3
PEO 4:	Demonstrate interpersonal skills, leadership ability and team building to achieve organization goals and serve society with professional ethics and integrity.	2	2	2	3

Note:.

Bloom's Taxonomy Knowledge Level	Knowledge Level Representation
Remember	K1
Understand	K2
Apply	K3
Analyse	K4
Evaluate	K5
Create	K6

Mapping / Correlation levels
1: Slight (Low)
2: Moderate (Medium)
3: Substantial (High)

PROGRAM STRUCTURE

I SEMESTER

Course Code	Course Title	Course Component	Course Type	Total Number of contact hours				Credits (C)
				Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
201HS1T01	Communicative English	HSC	Theory	3	0	0	3	3
201BS1T01	Differential Equations and Linear algebra	BSC	Theory	3	0	0	3	3
201BS1T04	Engineering Chemistry	BSC	Theory	3	0	0	3	3
201ES1T02	Programming for Problem Solving using C	ESC	Theory	3	0	0	3	3
201ES1I02	Computer Engineering Workshop	ESC	Integrated	2	0	2	4	3
201HS1L01	Communicative English Lab	HSC	Lab	0	0	3	3	1.5
201BS1L03	Engineering Chemistry Lab	BSC	Lab	0	0	3	3	1.5
201ES1L02	Programming for Problem Solving using C Lab	ESC	Lab	0	0	3	3	1.5
201MC1T01	Environmental Science	MC	Theory	2	0	0	2	0
			TOTAL	**16**	**0**	**11**	**27**	**19.5**

II SEMESTER

Course Code	Course Title	Course Component	Course Type	Total Number of contact hours				Credits (C)
				Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
201BS2T07	Numerical Methods and Complex Variables	BSC	Theory	3	0	0	3	3
201BS2T09	Applied Physics	BSC	Theory	3	0	0	3	3
201ES2T11	Computer Organization	ESC	Theory	3	0	0	3	3
201ES2T04	Python Programming	ESC	Theory	3	0	0	3	3
201ES2T07	Data Structures through C	ESC	Theory	3	0	0	3	3
201BS2L04	Applied Physics Lab	BSC	Lab	0	0	3	3	1.5
201ES2L06	Data Structures through C Lab	ESC	Lab	0	0	3	3	1.5
201ES2L14	Python Programming Lab	ESC	Lab	0	0	3	3	1.5
201MC2L01	Professional Communication Skills Lab	MC	Lab	0	0	3	3	0
201MC2T02	Constitution of India	MC	Theory	2	0	0	2	0
			TOTAL	**15**	**0**	**16**	**31**	**19.5**

III SEMESTER

Course Code	Course Title	Course Component	Course Type	Total Number of contact hours				Credits (C)
				Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
201CS3T01	Advanced Data Structures	PCC		3	0	0	3	3
201CS3T02	Object Oriented Programming through C++	PCC		3	0	0	3	3
201CS3T03	Operating Systems	PCC		3	0	0	3	3
201CS3T04	Software Engineering	PCC		3	0	0	3	3
201BS3T13	Discrete Mathematics	BSC		3	0	0	3	3
201CS3L01	Object Oriented Programming through C++ Lab	PCC		0	0	3	3	1.5
201CS3L02	Operating Systems Lab	PCC		0	0	3	3	1.5
201CS3L03	Unix and Shell Programming Lab	PCC		0	0	3	3	1.5
201SO3L05	Skill Oriented Course-I 1. Applications of python-numpy	SOC		0	0	4	4	2
201SO3L06	2.web application development using full stack frontend development–module –I							
201MC3T03	Biology for Engineers	MC		2	0	0	0	0
			TOTAL	**17**	**0**	**13**	**28**	**21.5**

IV SEMESTER

Course Code	Course Title	Course Component	Course Type	Total Number of contact hours				Credits (C)
				Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
	Probability and statistics	BSC		3	0	0	3	3
	Formal Languages and Automata Theory	PCC		3	0	0	3	3
	Database Management Systems	PCC		3	0	0	3	3
	Java programming	PCC		3	0	0	3	3
	Managerial Economics and Financial Accountancy	PCC		3	0	0	3	3
	Database Management Systems Lab	PCC		0	0	3	3	1.5
	Java Programming Lab	PCC		0	0	3	3	1.5
	R Programming Lab	PCC		0	0	3	3	1.5
	Skill Oriented Course (1. Applications of python-pandas 2. Web application development using full stack frontedn development module-II)	SOC		0	0	4	4	2
	Essence of Indian Traditional Knowledge	MC		2	0	0	0	0
			TOTAL	**17**	**0**	**13**	**28**	**21.5**

BSC: Basic Sciences Courses; HSMC: Humanities and Social Sciences including Management Courses; ESC: Engineering Sciences Courses; PCC: Professional Core Courses; SC: Skill Oriented Course; PEC: Professional Elective Courses; OEC: Open Elective Courses; MC: Mandatory Courses; PROJ: Project.

PROFESSIONAL ELECTIVE

Track		**Professional Elective I**	**Professional Elective II**	**Professional Elective III**	**Professional Elective IV**	**Professional Elective V**	**Professional Elective VI**
Track 1	**Programming languages**	Functional and Logic Programming	Scripting languages	C# .Net	Middleware Technologies	Design Patterns	Block chain Architecture Design and Use cases
Track 2	**Systems**	Advanced Computer Architecture	Advance Operating Systems	Distributed Systems	Embedded Systems	Fault Tolerant Computing	Real Time Operating Systems
Track 3	**Data Science and Machine Intelligence**	Artificial Intelligence	Machine Learning	Natural Language Processing	Data Analytics	No SQL databases	Deep Learning
Track 4	**Software Systems Engineering**	Software Requirement and Estimation	Software Testing Methodologies	Software Quality Assurance	Software Configuration Management	Agile Methodologies (Using Devops)	Software Project Management
Track 5	**Applications/ Generic**	Computer Graphics	Image Processing	Cloud Computing	Human Computer Interaction	Parallel Computing	Cyber Security

Sl. No.	**Course Work-Subject Area**	**Total no. of credits**	**no of credits as per APSCHE**	**no of credits as per AICTE**	**no of credits as per JNTUK**
1	Basic Sciences **(BSC)**	21	21	25	
2	Humanities and Social Sciences **(HSMC)**	10.5	10.5	12	
3	Engineering Sciences **(ESC)**	19.5	24	24	
4	Professional Core **(PCC)**	55.5	51	48	
5	Professional Electives **(PEC)**	15	15	18	
6	Open Electives **(OE)**	12	12	18	
7	Skill Oriented Course (SOC)	10	10	-	
8	Project **(PROJ)**	16.5	16.5	15	
9	Mandatory Non-Credit Courses (MC)	Non-credit	Non-credit	Non-credit	Non-credit
	Total	160	160	160	160

COMMUNICATIVE ENGLISH
(Common to all branches)

	L	T	P	C
I Semester	**L**	**T**	**P**	**C**
Course Code: 201HS1T01	**3**	**0**	**0**	**3**

Introduction:

In view of the growing importance of English as a tool for global communication and the consequent emphasis on training the students to acquire communicative competence, the syllabus has been designed to develop linguistic and communicative competence of the student's of Engineering.

As far as the detailed textbooks are concerned, the focus should be on the skills of listening, speaking, reading and writing. The non-detailed textbooks are meant for extensive reading for pleasure and profit.

Thus the stress in the syllabus is primarily on the development of communicative skills and fostering of ideas.

Course Objectives:

COB 1: To improve the language proficiency of the students in English with emphasis on LSRW skills.

COB 2: To enable the students to study and comprehend the prescribed lessons and subjects more effectively relating to their theoretical and practical components.

COB 3: To develop the communication skills of the students in both formal and informal situations.

COB 4: To develop the ways to overcome fear and use of words for irony.

COB 5: To make the learners understand the development conditions and the core Competences of the state to prioritize education system.

COB 6: To discuss that water is the world's most precious natural resources.

COB 7: To discuss how human sensitivity changes in accordance to times and situations in life.

COB 8: To inform the learner that all men can come together to abolish the war.

LISTENING SKILLS:
Objectives:

1. To enable the students to appreciate the role of listening skill and improve their pronunciation.
2. To enable the students to comprehend the speech of people belonging to different backgrounds and regions.
3. To enable the students to listen for general content, to fill up information and for specific information.

SPEAKING SKILLS:
Objectives:

1. To make the students aware of the importance of speaking for their personal and professional communication.
2. To enable the students to express themselves fluently and accurately in social and professional success.

3. To help the students describe objects, situations, and people.
4. To make the students participate in group activities like role-plays, discussionsand debates.
5. To make the students participate in just aminute talk.

READING SKILLS:
Objectives:

1. To enable the students to comprehend a text through silent reading.
2. To enable the students to guess the meanings of words, messages and inferences of texts in given contexts.
3. To enable the students to skim and scan a text.
4. To enable the students to identify the topic sentence.
5. To enable the students to identify discourse features.
6. To enable the students to make intensive and extensive reading.

WRITING SKILLS:
Objectives:

1. To make the students understand that writing is an exact formal skill.
2. To enable the students to write sentences, paragraphs, e-mails and essays.
3. To make the students identify and use appropriate vocabulary.
4. To enable the students to narrate and describe.
5. To enable the students to write coherently and cohesively.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Identify the ways to overcome fear and use of words for irony.
CO 2: Interpret the development conditions and the core competences of the state toprioritize education system.
CO 3: Explain water as the world's most precious natural resources.
CO 4: Illustrate human sensitivity to the changing times and situations in life.
CO 5: Identify that all men can come together to abolish the war.
CO 6: Rephrase coherent writing in social, political and religious background.
CO 7: Demonstrate writing and concepts of grammar skills.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	-	-	-	-	-	-	-	-	-	2	-	-
CO2 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO3 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO4 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO5 (K3)	-	-	-	-	-	-	-	-	-	2	-	-
CO6 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO7 (K2)	-	-	-	-	-	-	-	-	-	3	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K3)	-	-	-
CO2 (K2)	-	-	-
CO3 (K2)	-	-	-
CO4 (K2)	-	-	-
CO5 (K3)	-	-	-
CO6 (K2)	-	-	-
CO7 (K2)	-	-	-

Methodology:

1. The class is to be learner-centered where the learners are to read the texts to get a comprehensive idea of those texts on their own with the help of the peer group and the teacher.
2. Integrated skill development methodology must be adopted with focus on individual language skills as per the tasks/exercise.
3. The tasks/exercises at the end of each unit should be completed by the learners only and the teacher intervention is permitted as per the complexity of the task/exercise.
4. The teacher is expected to use supplementary material wherever necessary and also generate activities/tasks as per the requirement.
5. The teacher is permitted to use lecture method when a completely newConcept is introduced in the class.

Recommended Topics:

UNIT-I:

1. An Astrologers's Day - R.K.Narayan (Detailed)
2. Bade Bhai Saab – Munshi Premchand (Non-Detail)

UNIT-II:

1. Building A New State - A. P. J.Abdul Kalam
2. Morning Bells- Jayashree Mohan Raj (Non-Detail)

UNIT-III:

1. Water: The Elixir Of Life- C. V. Raman (Detailed)
2. The Power Of Plate Of Rice- Ifeoma Okoye (Non-Detail)

UNIT-IV:

1. The Woodrose-Abburi Chaya Devi (Detailed)
2. The Cop And The Anthem- O.Henry (Non-Detail)

UNIT-V:

1. Progress- St. John Ervine (Detailed)
2. Dial 000- Barry Rosenberg (Non-Detail)

Text Books:
Detailed Text Book: 'Using English' by Orient Black Swan.
Non Detailed Text Book: 'Life, language and Culture -Explorations' by Cengage.

Reference Books:

1. Objective English, Pearson Publications.
2. Effective English Communication, Tata Mc Graw-Hill Publishing.
3. Effective Technical English, Scitech.

Web Links:

1. http://sittingbee.com/an-astrologers-day-r-k-narayan/
2. http://bbrenglishforall.blogspot.com/2014/01/building-new-state-study-material.html
3. https://www.literatureworms.com/2012/10/water-elixir-of-life-by-sircvraman.html
4. http://macon.hol.es/woodrose-abburi-chaya-devi.pdf
5. https://ardhendude.blogspot.com/2013/07/analysis-of-progress-by-st-john-ervine.html

DIFFERENTIAL EQUATIONS AND LINEAR ALGEBRA
(Common to all branches)

I Semester	**L T P C**
Course Code: 201BS1T01	**3 0 0 3**

Course Objectives:

COB 1: To equip the students with the necessary mathematical skills and techniques that are essential for an engineering course.

COB 2: To help the student form a necessary base to develop analytical and design skills.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Apply the concepts of Mean Value theorem, Partial Differentiation and identify the maxima and minima of a given function.

CO 2: Solve the linear differential equations and model various situations involving differential equations of first order.

CO 3: Solve linear differential equations of higher order and model various situations involving second order differential equations.

CO 4: Calculate Rank of a matrix and solve the system of Linear equations and find the Eigen values and Eigen vectors.

CO 5: Compute various powers of a matrix and identify the nature of the quadratic form.

Mapping of course outcomes with program outcomes:

CO / PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO 1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 2 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 3 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 4 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 5 (K3)	3	2	-	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO 1 (K3)	-	-	-
CO 2 (K3)	-	-	-
CO 3 (K3)	-	-	-
CO 4 (K3)	-	-	-
CO 5 (K3)	1	1	1

UNIT-I:

Differential Calculus:

Rolle’s theorem, Lagrange’s theorem, Cauchy Mean Value theorem, Taylor’s and Maclaurin’s theorems (All theorems Without Proof).

Partial Differentiation: Euler’s theorem (without proof), Total derivative, Chain rule, Taylor’s and Maclaurin’s series expansion of functions of two variables, Jacobian, Functional dependence.

Applications:

Maxima and Minima of functions of several variables without constraints and with constraints (Lagrange’s method).

** (MATLAB Exercise: Plot graphs of various single and multivariable functions).

UNIT-II:

Differential Equations of First Order:

Introduction to differential equations, linear differential equation of first order - Bernoulli differential equation - Exact differential equations- Equations reducible to exact,

Applications:

Orthogonal trajectories, Newton’s Law of cooling, RL circuit.

UNIT-III:

Linear Differential Equations of Second And Higher Order:

Linear differential equations of higher order with constant coefficients, Complementary function and Particular integral with RHS term of the type polynomials in x, e^{ax}, sin ax,

cos ax, $e^{ax}V(x)$, xV(x)- Method of Variation of parameters, Equations reducible to constant coefficients –Cauchy-Euler equation, Legendre’s equation.

Application: LCR Circuit

** (MATLAB Exercise: Introduction to MATLAB commands and Solution of Initial Value Problems)

UNIT-IV:

System of Linear Equations, Eigen Values and Eigen Vectors:

Definition of a Vector space, Linear dependence and independence of vectors, Rank of a matrix, Echelon form and Normal form, solving system of Homogenous and Non homogenous linear equations- Gauss Jordon elimination method, Eigen values, Eigen vectors, Properties of eigen values and eigen vectors (without proof).

Applications:

Free vibrations of a two mass system.

UNIT-V:

Quadratic Forms:

Cayley -Hamilton theorem (without proof), Inverse and powers of a matrix by using Cayley - Hamilton theorem, Diagonalization of a matrix, Quadratic forms, Reduction of quadratic form to canonical form using orthogonal transformation, Nature of the quadratic form.

****(MATLAB Exercise:** Basic Operations on matrices, computation of rank, computation of eigen values and eigenvectors)

**Not to be examined

Text Books:

1. Advanced Engineering Mathematics, R.K.Jain, S.R.K.Iyenkar, Alpha Science Publications
2. Advanced Engineering Mathematics, Erwin Kreyszig, 10thEdition, Wiley-India.
3. Higher Engineering Mathematics, B.S.Grewal, 43rdEdition, KhannaPublishers.
4. Engineering Mathematics, P.Sivaramakrishna Das, C.Vijayakumari, Pearson Publications.

Reference Books:

1. Advanced Engineering Mathematics, D.G.Zill, MICHAIL R CULTER, 3rd Edition Norosa Publications2009.
2. Advanced engineering mathematics with MATLAB, Dean G. Duffy, CRCPress.

3. Advanced Engineering Mathematics, Peter O’neil, CengageLearning.
4. Advanced modern engineering mathematics, Glyn James, Pearson education.

Web Links:

1. https://nptel.ac.in/courses/111106100/
2. https://nptel.ac.in/courses/122107037/14
3. https://nptel.ac.in/courses/111106051/
4. http://mathworld.wolfram.com
5. https://www.khanacademy.org
6. https://spoken-tutorial.org/tutorial-search/?search_foss= MATLAB& search_ language=English

****

ENGINEERING CHEMISTRY
(Common to ECE, CSE& IT)

	L	T	P	C
I Semester	**L**	**T**	**P**	**C**
Course Code: 201BS1T04	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To acquaint the students with soft and hard water types and softening methods.
COB 2: To impart knowledge on the basic concepts of electrochemical cells and battery technology.
COB 3: To impart knowledge about polymers and plastic materials.
COB 4: Familiarize various sources of Non-renewable and renewable energy and their harnessing.
COB 5: To introduce different types of Nano-materials and the necessity of green chemistry.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Compare the quality of drinking water and problems associated with hard water.
CO 2: Outline the difference between primary and secondary cells.
CO 3: Explain different types of polymers and their applications.
CO 4: Compare various sources of Non-renewable and renewable energy.
CO 5: Summarize the importance of Nano materials and Green chemistry.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	2	-	-	-	-	-	-	-	-	-	-	-
CO2 (K2)	2	-	-	-	-	-	-	-	-	-	-	-
CO3 (K2)	2	-	-	-	-	-	-	-	-	-	-	-
CO4 (K2)	2	-	-	-	-	-	-	-	-	-	-	-
CO5 (K2)	2	-	-	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K2)	-	-	1
CO2 (K2)	1	-	-
CO3 (K2)	-	-	1
CO4 (K2)	-	-	1
CO5 (K2)	-	-	1

UNIT-I:

Introduction –Soft Water and hardness of water, types of hardness of water, degree of hardness of water, Units of hardness of water, problems on hardness, Boiler troubles - scale and sludge, Boiler corrosion, Industrial water treatment- zeolite and ion-exchange processes.

Specifications for drinking water, Bureau of Indian Standards (BIS) and World health organization(WHO) standards, - desalination of brackish water, reverse osmosis (RO) and electro dialysis.

UNIT-II:

Electrochemical Energy Systems:

Introduction-Electrochemical Cell(Galvanic cell), Electrochemical series, Applications,single electrode potential,Hydrogen and Calomel electrode,Nernst Equation for a single electrode ,Concentration Cells(Electrode & Electrolyte),Construction of glass electrode.

Batteries – Classical batteries-dry/Leclanche cell,Modern batteries-zinc air, lithium cells-Li MnO_2 cell- challenges of battery technology.Fuel cells- Introduction-classification of fuel cells – hydrogen and oxygen fuel cell, propane and oxygen fuel cell-Merits of fuel cell.

UNIT-III:

Polymer Chemistry:

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization, coordination polymerization, copolymerization (stereospecific polymerization) with specific examples and mechanisms of polymer formation (Free radical mechanism for addition polymerization)

Plastics - Thermoplastics and Thermosetting, Preparation, properties and applications of – PE, PVC, Bakelite, Teflon and Nylon-6, 6.

Elastomers–Buna-S, Buna-N–preparation, properties and applications.

Conducting polymers – polyacetylene, polyaniline – mechanism of conduction and applications.

UNIT-IV:

Energy Sources And Applications:

Introduction- sources of renewable energy –Hydro power, Biomass and Biofuels

Solar energy – Introduction -Physical and Chemical properties of Silicon- Preparation of Semiconductors - Doping of Silicon-p and n type semiconductors- PV cell / solar cell- Working &Manufacturing of Photovoltaic Cells using Chemical Vapor Deposition Technique-applications of solar energy.

Fuels: Introduction- classification- liquid fuels- Refining of petroleum-cracking-Reforming-Gaseous fuels-LPG & CNG Applications.

UNIT-V:

MATERIAL SCIENCE AND ENGINEERING:

Nanomaterial: Introduction to nanomaterial: nanoparticles, Nano cluster, carbon nanotube (CNT) and nanowires. Chemical synthesis of nanomaterial's: sol-gel method. Characterization: Principle and applications of scanning electron microscope (SEM) and transmission electron microscope (TEM).

Nanotubes: Carbon Nano tubes- Types of CNT's-preparation methods –Arc discharge, Laser abiliation and chemical vapour deposition –properties and applications.

Green Chemistry: Introduction, principles of green chemistry
(Ex: Solvent, Catalyst, Reactant)

BAND THEORY OF SOLIDS: Introduction –Explanation of conductors, semiconductors, Insulators by Band Theory- Super conductors-Types-Preparation-Propertiesand Applications**.**

APPENDIX: Introduction to Molecular Machines and Molecular Switches.

Text Books:

1. Engineering Chemistry, P.C. Jain and M. Jain, 16/e, Dhanapat Rai & Sons, (2014).
2. Engineering Chemistry, B.K. Sharma, KrishnaPrakasham, (2014).
3. Engineering Chemistry, Shikha Agarwal; Cambridge University Press, 2015 edition.

Reference Books:

1. A Textbook of Engineering Chemistry, Sashi Chawla, Dhanapath Rai and sons, (2003).
2. A Text Book of Nano Science and Nanotechnology, B.S Murthy and P. Shankar, University Press (2013).
3. A Textbook of Engineering Chemistry, S.S. Dara, S.Chand& Co, (2010).
4. Material Science and Engineering, V.Raghavan, Prentice-Hall India Ltd, (2004).
5. A text book of Engineering Chemistry, N.Krishna Murthy and Anuradha, Murthy Publications (2014).
6. Engineering Chemistry, K. SeshaMaheshwaramma and MridulaChugh, Pearson.

Web Links:

1. http://www.nptelvideos.in/2012/11/chemistry-of-materials
2. http://www.nptelvideos.com/lecture.php?id=2946
3. http://www.nptelvideos.com/lecture.php?id=2922
4. http://www.nptelvideos.com/lecture.php?id=2954

****

PROGRAMMING FOR PROBLEM SOLVING USING C

(Common to EEE,ECE,CSE & IT)

		L	T	P	C
I Semester					
Course Code:	**201ES1T02**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To impart adequate knowledge on the need of programming languages and problem solving techniques and develop programming skills.

COB 2: To enable effective usage of control structures and implement different operations on arrays.

COB 3: To demonstrate the use of strings and functions.

COB 4: To impart the knowledge on pointers and understand the principles of dynamic memory allocation.

COB 5: To make the students understand the concepts of structures, unions, files and their operations.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Illustrate the fundamental concepts of computers and basics of computer programming

CO 2: Make use of control structures and arrays in solving complex problems.

CO 3: Develop program on modular and strings fundamentals.

CO 4: Demonstrate the ideas of pointers usage.

CO 5: Solve real world problems using the concept of structures, unions and File operations.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1(K2)	2	1	-	-	1	-	-	-	-	-	-	3
CO2(K3)	3	2	1	1	3	-	-	-	-	-	-	3
CO3(K3)	3	2	1	1	3	-	-	-	-	-	-	3
CO4(K2)	2	1	-	-	2	-	-	-	-	-	-	3
CO5(K3)	3	2	1	1	3	-	-	-	-	-	-	3

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1(K2)	2	2	1
CO2(K3)	3	3	2
CO3(K3)	3	3	2
CO4(K2)	2	2	1
CO5(K3)	3	3	2

UNIT-I:

Computer History, Hardware, Software, Programming Languages and Algorithms:

Components and functions of a Computer System, Concept of Hardware and Software Programming Languages: Low-level and High-level Languages, Program Design Tools: Algorithm, Flowchart, Pseudo code.

Introduction to C Programming: Introduction, Structure of a C Program, Comments, Keywords, Identifiers, Data Types, Variables, Constants, Input/Output Statements, Operators, Type Conversion.

UNIT-II:
Control Flow, Relational Expressions & Arrays: Conditional Branching Statements if, if-else, if-else–if, switch. Basic Loop Structures: while, do-while loops, for loop nested loops, The Break and Continue Statements, goto statement.
Arrays: Introduction, Operations on Arrays, One dimensional Array, Two dimensiona Array, Multi dimensional arrays.

UNIT-III:
Strings: String Fundamentals, String Processing with and without Library Functions.
Functions: Introduction, Function Declaration, Function Definition, Function call Categories of Functions, passing parameters to Functions, Arrays as Functior Arguments, Scope of Variables, Variable Storage Classes,Recursion.

UNIT-IV:
Pointers: Concept of a Pointer, Declaring and Initializing Pointer Variables, Pointe Expressions and Address Arithmetic, Null Pointers, Generic Pointers, Pointers as Function arguments, Pointers and Arrays, Pointers and Strings, Pointer to Pointer Dynamic Memory Allocation, Dangling Pointer, Command line Arguments.

UNIT-V:
Structures, Unions, Bit Fields: Introduction, Nested Structures, Arrays of Structures Structures and Functions, Self-Referential Structures, Unions, Enumerated Data Type – enum variables, Using Typedef keyword, Bit Fields.
Data Files: Introduction to Files, Using Files in C, Reading from Text Files, Writing tc Text Files, Random File Access.

Text Books:

1. Computer Programming, Reema Thareja, Oxford University Press.
2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

Reference Books:

1. C Programming – A Problem Solving Approach, Forouzan, Gilberg, Cengage.
2. The C Programming Language, Dennis Richie And Brian Kernighan, Pearson Education.
3. Programming in C, Ashok Kamthane, 2ndEdition, Pearson Education.
4. Programming in ANSI C, E. Balagurusamy, 4E, Tata Mc Graw-Hill Education, 2008.

Web Links:

1. http://www.c4learn.com/
2. http://www.geeksforgeeks.org/c/
3. http://nptel.ac.in/courses/122104019/
4. http://www.learn-c.org/
5. https://www.tutorialspoint.com/cprogramming/

****

COMPUTER ENGINEERING WORKSHOP
(Common to CSE & IT)

		L	T	P	C
I Semester					
Course Code:	**201ES1I02**	**2**	**0**	**2**	**3**

Course Objectives:

COB 1: To make the students aware of the basic hardware components of a computer, installation of Linux Operating system and basic commands.
COB 2: To impart adequate knowledge on Virtual machine.
COB 3: To illustrate the usage of network and internet.
COB 4: To impart knowledge on IoT.
COB 5: To demonstrate the use of productivity & Office tools.
COB 6: To make the students prepare presentation/document using LaTeX.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Identify the components of a PC, Assemble & disassemble the same.
CO 2: Experiment with installation of Linux Operating System, Virtual machine and secure a computer from cyberthreats.
CO 3: Summarize the fundamentals and architecture of IoT.
CO 4: Prepare word documents; excel sheets and power point presentation.
CO 5: Develop presentation /documentation using Office tools and LaTeX.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	1	-	-	-	-	-	-	-	-	-	-
CO2 (K3)	3	2	1	-	3	-	-	-	-	-	-	-
CO3 (K2)	3	-	1	1	3	-	-	-	-	-	-	-
CO4 (K3)	2	1	1	-	3	-	-	-	-	-	-	-
CO5 (K3)	3	2	1	1	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	3	3	2
CO2 (K3)	3	3	2
CO3 (K2)	3	3	2
CO4 (K3)	3	3	2
CO5 (K3)	3	3	2

THEORY

UNIT-I:

Computer Hardware: Identification of peripherals of a computer and block diagram of a computer, I/O devices,I/O ports and interfaces,main memory,cache memory and secondary storage technologies,digital storage basics,networking components and speeds.

UNIT-II:

Operating System:

Setting up and configuring a new Virtual Machine, Setting up and configuring an existing Virtual Machine, Exporting and packaging an existing Virtual Machine into a portable format**.**

Linux operating system commands:
General command syntax, Basic help commands, Basic File system commands, Date and Time, Basic Filters and Text processing, Basic File compression commands, Miscellaneous: apt-get, vi editor**.**

UNIT-III:
Networking and Internet: Networking Commands: o ping, ssh, ifconfig, scp, netstat, ipstat, nslookup, traceroute, telnet, host, ftp, arp, wget, route,
Internet Services: Web Browser usage and advanced settings like LAN, proxy, content, privacy, security, cookies, extensions/ plugins , Antivirus installation, configuring a firewall, blocking pop-ups , Email creation and usage, Creating a Digital Profile on LinkedIn,

UNIT-IV:
Productivity tools: HTML tags, Introduction to HTML5 and its tags, Introduction to CSS3 and its properties.

UNIT-V:
Internet of Things (IoT):
IoT fundamentals, applications, protocols, communication models, architecture, IoT devices.

LIST OF PRACTICAL EXPERIMENTS

1. Identification of peripherals of a computer

1. Block diagram of the CPU along with the configuration of the each peripheral and its functions.

2. System Assembling and Disassembling

2.1) Disassembling the components of a PC.
2.2) Assembling the components back to working condition.

3. Virtual Machine setup

3.1) Setting up and configuring a new virtual machine.
3.2) Setting up and configuring an existing virtual machine.
3.3) Exporting and packaging an existing virtual machine into a portable format.

4. Installation of software's and basic commands

4.1) Installation of Linux operating Systems.
4.2) Basic Linux Operating System commands.

5. Networking and Internet

5.1) Networking commands.
5.2) Configuring Proxy and Firewall settings.
5.3) Exploring Internet and World Wide Web.
5.4) Exploring Search Engines, Cyber hygiene.

6. Productivity Tools-I

6.1) Basic HTML tags
6.2) Introduction to HTML5 and its tags

7. Productivity Tools-II

1. Introduction to CSS3 and its properties.

8. Productivity Tools-III

1. Preparation of a simple homepage

9. IoT

1. A study experiment on IoT fundamentals, applications, protocols, communication models, architecture, IoT devices

10. Office tools-I

1. Demonstration and Practice on Text Editors like Notepad++, Sublime Text, Atom, Brackets, Visual code, etc

11. Office tools-II

Demonstration and practice on

i. Microsoft Word- Formatting, Page Borders, Reviewing, Equations, symbols.
ii. Power Point- Features of power point, guidelines for preparing an effective Presentation.
iii. Microsoft Excel- Organize data, usage of formula, graphs and charts.

12. Office tools-III

12.1) Installation of LaTeX and related Software's.
12.2) Basic formatting using LaTeX.
12.3) Handling the equations in LaTeX.
12.4) Inserting the Tables in LaTeX.

List of Augmented Experiments:

(Any 2 of the following experiments can be performed)

13. Develop the web site for Hospital Management System
14. Prepare a power point presentation for college information (Include 10 slides)
15. List the common computer hardware problem and write down the solutions.
16. Prepare your Resume using LaTeX.

Text Books:

1. Computer Fundamentals, Anita Goel, Pearson Education, 2017
2. LATEX- User's Guide and Reference manual, Leslie Lamport, Pearson, LPE, 2/e.

Reference Books:

1. Essential Computer and IT Fundamentals for Engineering and ScienceStudents, Dr.N.B.Vekateswarlu, S.Chand**.**

Web Links:

1. https://assembleyourpc.net
2. https://www.latex-tutorial.com/tutorials
3. http://www.teachmsoffice.com
4. https://www.coursera.org/specializations/iot

COMMUNICATIVE ENGLISH LAB
(Common to all branches)

I Semester		**L T P C**
Course Code:	**201HS1L01**	**0 0 3 1.5**

Course Objectives:

COB 1: To facilitate computer-aidedmulti-media instruction enabling individual -ized and independent language learning.

COB 2: To sensitize the students to the nuances of English speech sounds, word accent and intonation.

COB 3: To bring a bout a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking.

COB 4: To improve the fluency in spoken English and neutralize mothertongue influence.

COB 5: To train students to use language appropriately.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Demonstrate nuances of language through audio-visual experience and group activities.

CO 2: Identify accent for intelligibility.

CO 3: Demonstrate in conversation, JAMs and public speaking.

CO 4: Make use of the concepts to communicate confidently and competently in English Language in all spheres.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO2 (K3)	-	-	-	-	-	-	-	-	-	2	-	-
CO3 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO4 (K3)	-	-	-	-	-	-	-	-	-	2	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K2)	-	-	-
CO2 (K3)	-	-	-
CO3 (K2)	-	-	-
CO4 (K3)	-	-	-

PRACTICE 1:

A. Greeting, Introducing and taking leave.

B. Pure Vowel.

PRACTICE 2:

A. Giving Information and Asking for Information.

B. Diphthongs.

PRACTICE 3:

A. Inviting, Accepting and Declining Invitations.

B. Consonants.

PRACTICE 4:

A. Commands, Instructions and Requests.
B. Accent and Rhythm.

PRACTICE 5:

A. Suggestions and Opinions.
B. Intonation.

Reference Books:

1. Strengthen your Communication Skills by Dr.M.Hari Prasad, Dr.SalivendraJ.Raju and Dr.G.Suvarna Lakshmi, Maruthi Publications.
2. English for Professionals by Prof Eliah, B.S Publications, Hyderabad.
3. A Hand book of English for Professionals by Prof Eliah, B.S Publications.
4. Effective Technical Communication by M.Ashraf Rizvi, Tata McGraw–Hill Publishing Company.
5. Word power made handy, Dr. Shalini verma, S.Chand Company.
6. Let us hear them speak, Jayashree Mohan raj, Sage texts.

Web Links:

1. https://fauzigeneraloflostsaga.wordpress.com/b-inggris/bab-2/
2. https://www.lawlessenglish.com/learn-english/pronunciation/vowels-phonetics/
3. https://www.english-at-home.com/giving-personal-information/
4. https://englishpost.org/make-accept-decline-invitations/
5. https://www.ef.com/ca/english-resources/english-grammar/reported-speech-orders-requests-suggestions/

****

ENGINEERING CHEMISTRY LAB

(Common to ECE, CSE & IT)

		L	T	P	C
I Semester					
Course Code:	**201BS1L03**	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To familiarize the students with the basic concepts of Engineering Chemistry lab.

COB 2: To demonstrate the digital and instrumental methods of analysis.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Calculate the hardness of water.

CO 2: Demonstrate Acid – Base titrations by instrumental analysis.

CO 3: Prepare advanced polymer materials

CO 4: Prepare alternative fuel like Bio-Diesel.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	-	-	-	3	-	-	-	-	-	-	-
CO2 (K3)	3	-	-	-	3	-	-	-	-	-	-	-
CO3 (K3)	3	-	-	-	3	-	-	-	-	-	-	-
CO4 (K3)	3	-	-	-	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K3)	-	-	1
CO2 (K3)	-	-	-
CO3 (K3)	-	-	-
CO4 (K3)	1	-	-

Introduction to Chemistry laboratory – Molarity, Normality, Primary, secondary standard solutions, Volumetric titrations, Quantitative analysis, Qualitative analysis, etc.

List of Experiments:

1. Determination of Total Hardness of a water sample.
2. Determination of Dissolved Oxygen in Water Sample.
3. Determination of Zinc by Complexometric method
4. P^H metric titration of (i) strong acid vs. strong base.
5. Determination of Fe (II) in Mohr's salt by potentiometric method.
6. Potentiometry – Titration between strong acid – strong base
7. Conductometric titrations (Strong acid vs Strong base)
8. Preparation of Phenol- Formaldehyde resin
9. Preparation of Urea-Formaldehyde resin.
10. Determination of percentage Moisture content in a coal sample.
11. Preparation of bio diesel.

List of Augmented Experiments:

(Any two of the following experiments can be performed)

12. Determination of acid value and saponification value of a given lubricant.
13. Determination of viscosity of a liquid.
14. Estimation of Calcium in port land Cement.
15. Determination of Vitamin – C.

Reference Books:

1. A Textbook of Quantitative Analysis, Arthur J. Vogel.
2. Dr.JyotsnaCherukuris (2012) Laboratory Manual of engineering chemistry-II, VGS Techno Series.
3. Chemistry Practical Manual, Lorven Publications K. Mukkanti (2009). Practical Engineering Chemistry, B.S. Publication.

****

PROGRAMMING FOR PROBLEM SOLVING USING C LAB
(Common to EEE, ECE, CSE & IT)

		L	T	P	C
I Semester					
Course Code:	**201ES1L02**	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To impart knowledge on basic Linux commands, various Editors, Raptor.
COB 2: To make the students understand the concepts of C programming.
COB 3: To nurture the students on Control Structures and develop different operations on arrays.
COB 4: To enable the students to learn string fundamentals and modular programming constructs.
COB 5: To impart knowledge on dynamic memory allocation.
COB 6: To explain the concepts of Structure, Unions and files for solving various problems.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Develop the basic programs in C and draw the flowcharts using Raptor.
CO 2: Make use of conditional and iterative statements to solve real time scenarios in C.
CO 3: Apply the concept of arrays, modularity and strings to handle complex problems.
CO 4: Apply the dynamic memory allocation functions using pointers.
CO 5: Develop programs using structures, and Files.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	2	1	-	-	1	-	-	-	-	-	-	3
CO2 (K3)	3	2	1	1	3	-	-	-	-	-	-	3
CO3 (K3)	3	2	1	1	3	-	-	-	-	-	-	3
CO4 (K3)	2	1	-	-	3	-	-	-	-	-	-	3
CO5 (K3)	3	2	1	1	3	-	-	-	-	-	-	3

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K2)	2	2	1
CO2 (K3)	3	3	2
CO3 (K3)	3	3	2
CO4 (K3)	2	2	1
CO5 (K3)	3	3	2

List of Experiments:

1. Introduction to C Programming

1.1) Basic Linux Commands
1.2) Exposure to Turbo C, Vi, Emacs, Code Blocks IDE, Dev C++
1.3) Writing simple programs using printf(), scanf()

2. Raptor

2.1) Installation and Introduction to Raptor.

2.2) Draw a flow chart to find the Sum of 2 numbers.
2.3) Draw a flow chart to find Simple interest.

For the below experiments develop flow charts using Raptor and implement using C:

3. Basic Math

3.1) Convert Celsius to Fahrenheit and vice versa.
3.2) Find largest of three numbers using ternary operator.
3.3) Calculate area of a Triangle using Heron's formula.

4. Control Flow- I

4.1) Find Whether the Given Year is a Leap Year or not.
4.2) Find the roots of a Quadratic Equation.
4.3) Make a simple Calculator to Add, Subtract, Multiply or Divide Using switch…case.

5. Control Flow- II

5.1) Find Whether the Given Number is Prime number or not
5.2) Find Whether the Given Number is Armstrong Number or not.
5.3) Print Floyd Triangle.

6. Control Flow- III

6.1) Find the sum of individual digits of a positive integer.
6.2) Check whether given number is palindrome or not.
6.3) Read two numbers, x and n, and then compute the sum of the geometric progression $1+x+x^2 +x^3 +\ldots\ldots\ldots\ldots+x^n$.

7. Arrays

7.1) Search an element in the given array (Linear Search)
7.2) Perform matrix addition.
7.3) Perform matrix multiplication.

8. Strings

8.1) Implement string manipulation operations with library function.
copy
concatenate
length
compare
8.2) Implement string manipulation operations without library function.
copy
concatenate
length
compare
8.3) Verify whether the given string is a palindrome or not

9. Functions, Array & Pointers

9.1) Demonstrate parameter passing in Functions.
9.2) Find Fibonacci, Factorial of a number with Recursion and without Recursion.
9.3) Find the sum of given numbers with arrays and pointers.

10. Pointers

10.1) Perform Addition, Subtraction, Multiplication and Division of two numbers using Command line arguments.
10.2) Find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc () function.
10.3) Find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc () function.

11. Structures

11.1) Store Information of a book Using Structure
11.2) Add Two Complex Numbers by Passing Structure to a Function

12. Files

12.1) Open a file and to print the contents of the file on screen.
12.2) Copy content of one file to another file.
12.3) Merge two files and store content in another file.

LIST OF AUGMENTED EXPERIMENTS:

(Any 2 of the following experiments can be performed)

13. ATM Pin Generation:

Aditya purchased a credit card. He hasto generate a PIN number to access the ATM and Net banking for which OTP was sent to his registered mobile number. Using this OTP number, he has to generate ATM PIN number. After generating PIN number, he can use it for further transactions. Maximum login he can make is 3 times.

Sample Input:
OTP: 6732
If valid
Enter PIN: 8858
Confirm your PIN: 8858

Sample output:
valid/Invalid
PIN generated successfully.
Note: OTP is hard coded.

14. Reset Password:

Aditya was using Syndicate Bank’s Online Account. She wanted to pay her bills through Online. But she forgets her password. Now she has to reset the password. For resetting the password, she has to select reset option from the Menu.
NOTE: using switch case.

Sample input:
Fast withdrawal
Mini Statement
Balance Enquiry
Reset Password Enter your choice: 4
Sample Output:
Reset password: New password: ***** Confirm password: *****

15. Student Attendance Report Generation:

Some of the school staff had failed to maintain the attendance of the students, causing lack ofessential records related to student’s attendance that should be submitted in a parent meet. The school management has decided to automate the process in order to maintain the attendance of every student effectively. You are asked to write a program to the above scenario and display whether the student can write the Exam or not.

Percentage<65	Detained
>=65 and <75	should pay condonation to appear for Exams
>=75	Allowed for exams

Sample Input:
Enter no of students: 5
Enter Students Details:

Rno:1	Name: Kalyan	Attendance (%): 67	Should pay condonation to appear for exams
Rno:2	Name: Laxman	Attendance (%): 56	
Rno:3	Name: Yamini	Attendance (%): 79	
Rno:4	Name: Aryan	Attendance (%): 60	
Rno:5	Name: Raghav	Attendance (%): 88	

Sample output:

Rno	Name	Attendance (%)	Remarks
1	Kalyan	67	67 should pay condonation to appear for Exams
2	Laxman	56	detained
3	Yamini	79	allowed for Exams
4	Aryan	60	detained
5	Raghav	88	allowed for Exams

16. Library Management

Shilpa is a student of PGEC got the Library Card. She wants to lend the books from the Library. The college gave two cards to every student. The students can lend only two books at a time and it must be returned after 15 days. If the books are not returned, late fee will be collected for no. of days lagged, after the due date. Late fee per day is Rs.50/-

```
Sample output:
Enter the name of student: nalini
Enter the Roll No.:555
Enter the branch: cse
Enter the section: A
Enter the year: 3
Enter the Date of Lend (dd mm yyyy):  08 08 2017
Enter the Date of return (dd mm yyyy): 09 10 2017
The no. of days book used by the student = 62
```

```
Extra days used by the student = 47
Late return fine fee = 2350
```

Reference Books:

1. Let Us C YashwanthKanetkar, Eighth edition, BPB Publications.
2. Programming in C A-Practical Approach Ajay Mittal. Pearson Education.
3. The C programming Language, Dennis Richie and Brian Kernighan, Pearson Education.

Web Links:

1. https://www.hackerrank.com/
2. https://www.codechef.com/
3. https://www.topcoder.com/
4. https://code-cracker.github.io/
5. https://raptor.martincarlisle.com/
6. https://nptel.ac.in/courses/106105085/2

****

ENVIRONMENTAL SCIENCE
(Common to all branches)

I Semester		**L**	**T**	**P**	**C**
Course Code:	**201MC1T01**	**2**	**0**	**0**	**0**

Course Objectives:

COB 1: To define the various ecosystems and its diversity.
COB 2: To summarize the overall natural resources.
COB 3: To classify environmental impacts of developmental activities.
COB 4: To discuss social issues, environmental legislation and global t treaties.
COB 5: To educate human population and environment.

Course Outcomes:
At the end of the Course, Student will be able to:

CO 1: Identify the need for protecting the producers and consumers in various ecosystems and their role in the food web.
CO 2: Outline the natural resources and their importance for the sustenance of the life.
CO 3: List out the biodiversity of India, threats and its conservation methods.
CO 4: Illustrate various attributes of the pollution, impacts and measures to control the pollution along with waste management practices.
CO 5: Describe social issues both rural and urban environment to combat the challenges.
CO 6: Summarize the legislations of India in environmental protection.
CO 7: Explains the population growth and its implications.
CO 8: Transforms existing campus into self sustaining green campus with environment Friendly aspects of – Energy, Water and waste water reuse, plantation, rain water Harvesting and Parking Curriculum.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K1)	-	-	-	-	-	-	2	-	-	-	-	-
CO2 (K2)	-	-	-	-	-	-	3	-	-	-	-	-
CO3 (K1)	-	-	-	-	-	-	2	-	-	-	-	-
CO4 (K2)	-	-	-	-	-	-	3	-	-	-	-	-
CO5 (K2)	-	-	-	-	-	-	3	-	-	-	-	-
CO6 (K2)	-	-	-	-	-	-	3	-	-	-	-	-
CO7 (K4)	-	-	-	-	-	-	3	-	-	-	-	-
CO8 (K2)	-	-	-	-	-	-	3	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K1)	-	-	-
CO2 (K2)	-	2	-
CO3 (K1)	-	-	-
CO4 (K2)	-	-	-
CO5 (K2)	-	-	-
CO6 (K2)	-	-	-
CO7 (K4)	-	-	-
CO8 (K2)	2	2	-

UNIT-I:

Multidisciplinary Nature of Environmental Studies:

Definition, Scope and Importance, Need for Public Awareness.

Natural Resources: Renewable and non-renewable resources – Natural resources and associated problems.

UNIT-II:

Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers and decomposers. Food chains, food webs and ecologicalpyramids.

Biodiversity and Its Conservation: Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India – Values of biodiversity. Hot-spots of biodiversity – Threats to biodiversity. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT-III:

Environmental pollution and Solid Waste Management:

Environmental Pollution: Definition, Cause, effects and control measures of:

a. AirPollution.
b. Waterpollution
c. Soilpollution
d. Marinepollution
e. Noisepollution

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution.

UNIT-IV:

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to Energy &Water. Resettlement and rehabilitation of people, Environmental ethics, Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Environment Protection Act – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act-Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

UNIT-V:

Human Population and the Environment: Population growth, variation among nations. Environment and human health, Human Rights, Value Education. Role of Information Technology in Environment and human health.

Text Books:

1. Text book of Environmental Studies for Undergraduate Courses by ErachBharucha for University Grants Commission, UniversitiesPress.
2. Environmental Studies by Palaniswamy – Pearsoneducation.
3. Environmental Studies by Dr.S.AzeemUnnisa, Academic PublishingCompany.

Reference Books:

1. Textbook of Environmental Science by Deeksha Dave and E.Sai Baba Reddy, CengagePublications.
2. Text book of Environmental Sciences and Technology by M.Anji Reddy, BS Publication.

3. Comprehensive Environmental studies by J.P.Sharma, Laxmipublications.
4. Environmental sciences and engineering – J. Glynn Henry and Gary W. Heinke – Prentice hall of India Privatelimited.
5. A Text Book of Environmental Studies by G.R.Chatwal, Himalaya PublishingHouse.
6. Introduction to Environmental engineering and science by Gilbert M. Masters and Wendell P. Ela - Prentice hall of India Privatelimited.

Web Links:

1. https://www.youtube.com/watch?v=mOwyPENHhbc
2. https://www.youtube.com/watch?v=_mgvsPnCYj4
3. http://nptel.ac.in/courses/122102006/1-2

****

NUMERICAL METHODS AND COMPLEX VARIABLES

(Common to CSE & IT)

		L	T	P	C
II Semester					
Course Code:	**201BS2T07**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To equip the students with the necessary mathematical skills andtechniques that are essential for an engineering course.

COB 2: To help the students acquire a necessary base to develop analytical anddesign skills.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Apply numerical methods to obtain approximate solution of equations.

CO 2: Apply various numerical methods to interpolate polynomials.

CO 3: Apply numerical methods to initial value problems and problems involving integration.

CO 4: Identify the analyticity of functions of complex variables.

CO 5: Apply Cauchy's theorem, Cauchy's integral formula and Cauchy's residue theorem.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO2 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO3 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO4 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO5 (K3)	3	2	-	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO /PO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K3)	1	-	-
CO2 (K3)	1	-	-
CO3 (K3)	-	-	-
CO4 (K3)	-	-	-
CO5 (K3)	-	-	-

UNIT-I:

Solution of Algebraic and Transcendental Equations:

Introduction to Numerical methods, Bisection method, Secant method, Method of false position, Iteration method, Newton - Raphson method.

UNIT-II:

Interpolation:

Introduction to Interpolation, Finite differences, Forward differences, Backward differences, Relation between operators, Newton's formula for interpolation, Lagrange's interpolation, Newton's divided difference interpolation.

UNIT-III:

Numerical Integration and solution of Ordinary Differential equations:

Trapezoidal rule, Simpson's 1/3rdand 3/8thrule, Solution of ordinary differential equations by Taylor's series, Picard's method of successive approximations, Euler's method, Modified Euler's method, Runge - Kutta method (fourth order).

UNIT-IV:

Functions of Complex variables:

Introduction, Continuity, Differentiability, Analyticity, Properties of analytic functions, Cauchy-Riemann equations in Cartesian and polar co-ordinates, Harmonic functions, Milne Thompson method.

UNIT-V:

Complex Integration:

Introduction to complex integration, Cauchy's integral theorem, Cauchy integral formula, Liouville's theorem, Taylor's series, Maclaurin's series, Laurent's series (All theorems without proof), Singular point, Types of singularities-Isolated, Essential and Removable singularities, pole of order m, Residues, Cauchy Residue theorem.

Text Books:

1. Advanced Engineering Mathematics, R.K.Jain, S.R.K.Iyenkar, Alpha Science Publications.
2. Advanced Engineering Mathematics, Erwin Kreyszig, 10thEdition, Wiley-India.
3. Higher Engineering Mathematics, B.S.Grewal,43rdEdition, KhannaPublishers.
4. Engineering Mathematics, P.Sivaramakrishna Das, C.Vijayakumari, Pearson Publications.

Reference Books:

1. Advanced engineering mathematics with MATLAB, Dean G. Duffy, CRCPress.
2. Higher Engineering Mathematics, Dass H.K., Rajnish Verma. Er., S. Chand Co. Pvt. Ltd,Delhi.
3. Higher engineering mathematics by John Bird, 5thedition Elsevier Limited,2006.

Web Links:

1. https://nptel.ac.in/courses/111107108/25
2. https://nptel.ac.in/courses/111103021/
3. https://nptel.ac.in/courses/111107105/
4. http://mathworld.wolfram.com
5. https://www.khanacademy.org

****

APPLIED PHYSICS

(Common to ECE, CSE & IT)

		L	T	P	C
II Semester					
Course Code:	**201BS2T09**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To impart Knowledge of Physical Optics phenomena like Interference andDiffraction required to design instruments with higher resolution.

COB 2: To make the student understand Physics of Semiconductors and theirworking mechanism for their utility in sensors.

COB 3: To impart the knowledge of materials with characteristic utility inappliances.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Apply the principles of interference and diffraction to design and enhance the resolving power of various optical instruments.

CO 2: Explain the fundamental concepts of Quantum behavior of matter.

CO 3: Classify the solids based on energy band structure.

CO 4: Explain the basic concepts of Semi-Conductors and Identify the type ofsemiconductors using Hall Effect.

CO 5: Explain about magnetic and dielectric properties of different materials.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO2 (K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO3 (K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO4 (K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO5 (K3)	3	2	-	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	-	-	-
CO2 (K2)	-	1	-
CO3 (K2)	-	-	-
CO4 (K2)	-	-	-
CO5 (K3)	1	1	-

UNIT-I:

Wave Optics: Principle of Superposition - Interference of light - Conditions for sustained Interference - Interference in thin films (reflected geometry) - Newton’s Rings (reflected geometry). Diffraction – Fraunhofer Diffraction - Diffraction due to Single slit (quantitative), Doubleslit, N -slits and circular aperture (qualitative) – Intensity distribution curves - DiffractionGrating– Grating spectrum – missing order– resolving power – Rayleigh’s criterion – Resolving powers of Microscope, Telescope and grating (qualitative).

UNIT-II:

Quantum Mechanics: Introduction – Matter waves – de Broglie’s hypothesis – Davisson- Germer experiment – G.P.Thomson experiment – Heisenberg’s Uncertainity Principle – interpretation of wave function – Schröedinger Time Independent and Time Dependent wave equations – Particle in a potential box.

UNIT-III:

FreeElectronTheory: Introduction–Classicalfreeelectrontheory(meritsanddemeritsonly)-QuantumFreeelectrontheory–electricalconductivitybasedonquantumfreeelectrontheory– Fermi Dirac distribution function – Temperature dependence of Fermi-Dirac distribution function - expression for Fermi energy -Density of states.

BAND THEORY OF SOLIDS Bloch's theorem (qualitative) – Kronig-Penney model(qualitative) – energy bands in crystalline solids – E Vs K diagram – classification of crystalline solids – effective mass of electron – m^* Vs K diagram - concept of hole.

UNIT-IV:

Semiconductor Physics: Introduction– Intrinsic semi-conductors - density of charge carriersElectrical conductivity – Fermi level – extrinsic semiconductors - p-type & n-type - Density of charge carriers - Dependence of Fermi energy on carrier concentration and temperature – Hall effect- Hall coefficient - Applications of Hall effect - Drift and Diffusion currents – Einstein'sequation.

UNIT-V:

Magnetism: Introduction – Magnetic dipole moment – Magnetization – Magnetic susceptibility and permeability – Origin of permanent magnetic moment – Bohr magneton – Classification of magnetic materials: Dia, para & Ferro – Domain concept of FerromagnetismHysteresis – soft and hard magnetic materials – applications of Ferromagnetic material.

Dielectrics: Introduction-Dielecticpolarization–DielectricPolarizability,Susceptibilityand Dielectric constant-types of polarizations: Electronic and Ionic (Quantitative), Orientational polarizations (qualitative) – Lorentz Internal field – Claussius-Mossoti equation - Frequency dependence of polarization – Applications ofdielectrics.

Text Books:

1. " AText book of Engineering Physics" by M N Avadhanulu, P G Kshirsagar&T.V.S.Arun Murthy S Chand &Company Ltd, 11th edition.
2. "EngineeringPhysics"byD.K.BhattacharyaandPoonamTandon,Oxfordpress(2015).
3. "Engineering Physics" by R.K Gaur. and S.L Gupta., - Dhanpat Rai publishers,2012.

Reference Books:

1. "Engineering Physics" by M.R.Srinivasan, New Age international publishers(2009).
2. "Optics" by AjoyGhatak, 6th Edition McGraw Hill Education,2017.
3. Engineering Physics by Mani naidu – Peasrson Publications –2017.

Web Links:

1. http://nptel.ac.in/courses/122107035/11
2. http://nptel.ac.in/courses/115102023/
3. https://phet.colorado.edu/en/simulations/category/physics
4. http://physicsgecg.blogspot.in/p/reading-materials.html
5. https://sites.google.com/site/physicsbysureshsaganti/home

COMPUTER ORGANIZATION

(Common to CSE & IT)

		L	T	P	C
II Semester					
Course Code:	**201ES2T11**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To make the students learn the common forms of number systems and their conversions.

COB 2: To provide the knowledge on combinational and sequential logic circuits.

COB 3: To make the students understand the operation of CPUs including instruction cycle and buses.

COB 4: To make the students aware of the fundamentals of different instruction set architectures and their relationship to the CPU design.

COB 5: To describe Memory System and I/O Organization.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Recognize and manipulate representations of numbers stored in digital computers.

CO 2: Relate Postulates of Boolean algebra and minimize combinational functions.

CO 3: Make Use of combinational and sequential circuits with simplified logic functions in various digital circuits.

CO 4: Explain the Basic Computer Organization and Design

CO 5: Describe the internal organization of computers, CPU, memory unit and Input/Outputs and the relations between its main components.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K3)	3	2	-	-	3	-	-	-	-	-	-	-
CO2 (K2)	2	1	-	-	2	-	-	-	-	-	-	-
CO3 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO4 (K2)	2	1	-	-	2	-	-	-	-	-	-	-
CO5 (K2)	2	1	-	-	2	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	2	2	2
CO2 (K2)	3	3	1
CO3 (K3)	3	3	2
CO4 (K2)	2	2	1
CO5 (K2)	2	2	1

UNIT-I:

Digital Components and Data Representation: Introduction, Numbering Systems, Decimal to Binary Conversion, Binary Coded Decimal Numbers, Weighted Codes, Self-Complementing Codes, Cyclic Codes, Error Detecting Codes, Error Correcting Codes, Hamming Code for Error Correction, Alphanumeric Codes, ASCII Code.

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Boolean Algebra and Logical gates:

Boolean Algebra :Theorems and properties, Boolean functions, canonical and standard forms , minimization of Boolean functions using algebraic identities; Karnaugh map representation and minimization using two and three variable Maps ;Logical gates ,universal gates and Two-level realizations using gates : AND-OR, OR-AND, NAND-NAND and NOR-NOR structures.

UNIT-II:

Digital logic circuits: Combinatorial Circuits: Introduction, Combinatorial Circuit Design Procedure, Integrated NAND-NOR Gates, Multifunction gates, Multi-bit adder, Multiplexers, Demultiplexers, Decoders.

Sequential Switching Circuits: Latches and Flip-Flops, Ripple counters using T flipflops; Synchronous counters; Shift Registers; Ring counters.

UNIT-III:

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

Register Transfer language and microinstructions : Bus memory transfer, arithmetic and logical micro-operations, shift and rotate micro-operations

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT-IV:

Micro programmed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

UNIT-V:

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access.

Text Books:

1. Digital Logic and Computer Design, Morris Mano, Pearson Education,11th Edition.
2. Computer System Architecture, M.MorrisMano, PHI, 3rd edition.

Reference Books:

1. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI, 2006.
2. Computer Organization, Hamacher, Vranesic and Zaky,TMH, 5th edition, 2002
3. Computer Organization & Architecture: Designing for Performance, William Stallings, PHI, 7th edition,2006.

Web Links:

1. http://nptel.ac.in/courses/106106092/
2. http://nptel.ac.in/courses/106103068/2
3. http://www.cuc.ucc.ie/CS1101/David%20Tarnoff.pdf
4. https://www.geeksforgeeks.org/computer-arithmetic-set-1/
5. https://onlinecourses.nptel.ac.in/noc20_ee11/preview

****

PYTHON PROGRAMMING
(Common to CSE & IT)

	L	T	P	C
II Semester	**L**	**T**	**P**	**C**
Course Code: 201ES2T04	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To impart the knowledge on input, output statements, decision structures and operators in python

COB 2: To facilitate the students, apply control statements and string functions in Python Scripts.

COB 3: To facilitate the students, apply control statements and string functions in Python Scripts.

COB 4: To demonstrate the Object Oriented Concepts and file concepts.

COB 5: To enable the students develop GUI applications

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Develop programs using fundamental concepts in python

CO 2: Solve problems using control statements and string methods.

CO 3: Develop real time applications using data structures and functions

CO 4: Apply Object Oriented Programming concepts and files.

CO 5: Build various applications using GUI and exceptions.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K3)	3	2	1	-	3	-	-	3	-	-	3	-
CO2 (K3)	3	2	-	-	3	-	-	3	-	-	3	-
CO3 (K3)	3	2	1	-	3	-	-	3	-	-	3	-
CO4 (K3)	3	2	1	-	3	-	-	3	-	-	3	-
CO5 (K3)	3	2	1	-	3	-	-	3	-	-	3	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	3	-	2
CO2 (K3)	-	-	2
CO3 (K3)	3	3	2
CO4 (K3)	3	3	2
CO5 (K3)	3	3	2

UNIT-I:

Programming: Introduction to Programming Concepts with Scratch

Introduction: Introduction to Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, Operators. Type conversions, Expressions, More about Data Output. Data Types

Expression: Strings Assignment, and Comment, Numeric Data Types and Character Sets, Using functions and Modules.

Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops

UNIT-II:

Control Statement: Definite iteration for Loop Formatting Text for output, Selection if and if else Statement Conditional Iteration The While Loop

Strings and Text Files: Accessing Character and Substring in Strings, Data Encryption, Strings and Number Systems, String Methods Text Files.

UNIT-III:

List and Dictionaries: Lists, Defining Simple Functions, Dictionaries Design with

Function: Functions as Abstraction Mechanisms, Problem Solving with Top Down Design, Design with Recursive Functions, Case Study Gathering Information from a File System, Managing a Program’s Namespace, Higher Order Function.

Modules: Modules, Standard Modules, Packages

UNIT-IV:

Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance , overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using Oops support Design with Classes: Objects and Classes, Data modeling Examples, Case Study An ATM, Structuring Classes with Inheritance and Polymorphism

File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations

UNIT-V:

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Redefined Clean-up Actions.

Graphical User Interfaces: The Behavior of Terminal Based Programs and GUI - Based, Programs, Coding Simple GUI-Based Programs, Other Useful GUI Resources.

Text Books:

1. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage, 2018.
2. 1. Beginning Python: from Novice to Professional, Lie Hetland, Magnus, 2nd Edition, 2005.

Reference Books:

1. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
2. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson, 2017.
4. Think Python, Allen Downey, Green Tea Press,2012 .
5. Python for Everybody Exploring Data in Python 3, Charles Russell Severance, Sue

Blumenberg.

6. Learning Python, Mark Lutz, Orielly, 2013.

Web Links:

1. https://www.python.org
2. https://www.coursera.org/courses?query=Python%20programming
3. https://www.learnPython.org/
4. https://www.tutorialspoint.com/python3/python_tutorial.pdf
5. http://www.geeksforgeeks.org/Python

****

DATA STRUCTURES THROUGH C

(Common to EEE, CSE&IT)

		L	T	P	C
II Semester					
Course Code:	**201ES2T07**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To make the students learn the basic Concepts of Data Structures and algorithms.
COB 2: To enable the students analyze the various Searching and Sorting techniques.
COB 3: To provide knowledge on Stacks and Queues
COB 4: To impart knowledge on list and its applications.
COB 5: To facilitate the students learn the various operations of Trees.
COB 6: To demonstrate the Graph Traversal Techniques.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Illustrate time and space complexities of an algorithm.
CO 2: Apply various searching and sorting techniques to solve computing problems.
CO 3: Make use of linear data structures to solve real time problems.
CO 4: Develop applications using Tree Data Structures.
CO 5: Solve problems using Graph Algorithms.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO10 (K2)	PO11 (K3)	PO12 (K1)
CO1(K2)	2	1	-	-	2	-	-	-	-	-	-	-
CO2(K3)	3	2	-	-	3	-	-	-	-	-	-	-
CO3(K3)	3	2	-	-	3	-	-	-	-	-	-	-
CO4(K3)	3	2	-	-	3	-	-	-	-	-	-	-
CO5(K3)	3	2	-	-	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1(K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K2)	2	2	3
CO2 (K3)	3	-	3
CO3 (K3)	3	-	3
CO4 (K3)	3	3	3
CO5 (K3)	3	3	3

UNIT-I:

Data Structures –Definition, Classification and Operations on Data Structures, Pseudo code, Algorithm analysis, Time and Space Complexity.
Searching: Linear search, Binary search.
Sorting: Insertion Sort, Selection Sort, Exchange (Bubble Sort, Quick Sort),merging (Merge sort), distribution (Radix Sort) algorithms.

UNIT-II:

Stacks: Introduction, Array Representation of Stacks, Operations and Implementation, Applications of Stacks-Reversing list, Infix to Postfix Conversion, Evaluating Postfix Expressions.

Queues: Introduction, Array Representation of Queues, Operations and Implementation, Types of Queues: Circular Queues, Deques and Priority Queues, Application of Queues.

UNIT-III

Linked Lists: Introduction, Singly linked list, Operations on Singly Linked list - Insertion, Deletion and Searching, Doubly linked list - Insertion, Deletion, Circular linked list-Insertion, Deletion, Linked Representation of Stacks and Queues, Applications of Linked lists-Addition of Polynomials, Sparse Matrix Representation using Linked List.

UNIT-IV

Trees: Basic Terminology in Trees, Binary Trees-Properties, Representation of Binary Trees using Arrays and Linked lists, Traversing a Binary Tree(In-Order, Pre-Order,Post-Order).

Binary Search Trees: Definition, Operations: Searching, Insertion, Deletion, Applications-Expression Trees, Heap Sort, Balanced Binary Trees- AVL Trees, Insertion, Deletion and Rotations.

UNIT-V

Graphs: Introduction, Graph Terminology, Representation of Graphs-Adjacency Matrix and using Linked list, Graph Traversals(BFT & DFT),Applications-Minimum Spanning Tree Using Prims &Kruskals Algorithm, Dijkstra’s Shortest Path, Warshall’s Algorithm, Transitive Closure. (Algorithmic Concepts Only, No Programs required).

Text Books:

1. Data Structures Using C,Reema Thareja, Oxford University Press,2nd Edition.
2. Data Structures and Algorithm Analysis In C, Mark Allen Weiss,2nd Edition.

Reference Books:

1. Fundamentals of Data Structure in C, Horowitz,Sahni, Anderson Freed, University Press, 2nd Edition, 2008.
2. Data Structures, Richard F, Gilberg, Forouzan, Cengage Learning,2nd Edition.
3. Data Structures and Algorithms, G. A.V.Pai, TMH,2008.

Web Links:

1. http://nptel.ac.in/courses/106102064/
2. http://algs4.cs.princeton.edu/home/
3. https://faculty.washington.edu/jstraub/dsa/Master_2_7a.pdf
4. http://www.udacity.com/
5. http://www.courseera.com/

****

APPLIED PHYSICS LAB
(Common to ECE, CSE & IT)

		L	T	P	C
II Semester		**L**	**T**	**P**	**C**
Course Code:	**201BS2L04**	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To make the students gain practical knowledge to co- relate with thetheoretical studies.
COB 2: To impart skills in measurements.
COB 3: To plan the experimental procedure and to record and process theresults.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Use spectrometer, travelling microscope for making measurements.
CO 2: Determine energy gap of a semiconductor, draw characteristic curvestoestimate thermal coefficient of a thermistor, Zener diode.
CO 3: Determine the dielectric constant and resistivity.
CO 4: Determine wavelength of source and width of the narrow slits.
CO 5: Find the strength of magnetic field.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1(K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO2(K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO3(K3)	3	1	-	-	-	-	-	-	-	-	-	-
CO4(K2)	2	2	-	-	-	-	-	-	-	-	-	-
CO5(K2)	2	2	-	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K2)	-	-	-
CO2 (K3)	-	-	-
CO3 (K3)	-	-	-
CO4 (K2)	1	1	-
CO5 (K2)	1	1	-

LIST OF EXPERIMENTS

(Any 10 of the following listed experiments)

1. Determination of wavelength of a source-Diffraction Grating-Normalincidence.
2. Newton’s rings – Radius of Curvature of Plano - ConvexLens.
3. Determination of thickness of a spacer using wedge film and parallel interferencefringes.
4. Magnetic field along the axis of a current carrying coil – Stewart and Gee’sapparatus.
5. Energy Band gap of a Semiconductor p - njunction.
6. Characteristics of Thermistor – TemperatureCoefficients.
7. Determination of dielectric constant by charging and dischargingmethod.
8. Determination of resistivity of semiconductor by Four probemethod.
9. Study the variation of B versus H by magnetizing the magnetic material (B-Hcurve).

10. Measurement of magnetic susceptibility by Quincke’smethod.
11. Dispersive power of diffractiongrating.
12. Resolving Power oftelescope.
13. Resolving power ofgrating.
14. Determination of Hall voltage and Hall coefficients of a given semiconductor usingHall effect.
15. Variation of dielectric constant withtemperature.

LIST OF AUGMENTED EXPERIMENTS

(Any two of the following experiments can be performed)

16. Determine the Young's Modulus of the material of the bar subjected to uniformbending.
17. Determine the Young's Modulus of the material of the bar subjected to non-uniformbending.
18. V-I characteristics of P-N junctionDiode.
19. V-I characteristics and Breakdown voltage of ZenerDiode.

Reference Books:

1. Engineering Physics Lab Manual by Dr.C.V.Madhusudhana Rao, V.VasanthKumar, ScitechPublications.
2. Laboratory Manual Cum Record for Engineering Physics I & II by Dr.Y.Aparna, Dr.K.Venkateswara Rao, VGSTechnoseries.

APPLIED PHYSICS - VIRTUAL LAB – ASSIGNMENTS

LIST OF EXPERIMENTS

1. HallEffect
2. CrystalStructure
3. Brewster’sangle
4. Numerical Aperture of Opticalfiber
5. PhotoelectricEffect
6. LASER – Beam Divergence and Spotsize
7. Michelson’sinterferometer
8. Black bodyradiation
9. Flywheel –moment ofinertia
10. AC Sonometer
11. Resistivity by four probemethod
12. Newton’s rings –Refractive index ofliquid

Web Links:

URL: www.vlab.co.in

****

DATA STRUCTURES THROUGH C LAB

(Common to EEE, CSE& IT)

	L	T	P	C
II Semester	**L**	**T**	**P**	**C**
Course Code: 201ES2L06	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To impart knowledge on linear and non-linear data structures.
COB 2: To facilitate the students, learn different sorting and searching techniques.
COB 3: To make the students implement operations and applications of stacks and queues.
COB 4: To demonstrate various operations on linked lists.
COB 5: To provide knowledge on trees and its operations.
COB 6: To illustrate concept of graph representation and traversals.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Develop programs using recursive functions.
CO 2: Examine searching, sorting algorithms.
CO 3: Developprograms for implementing various operations on linear data structures.
CO 4: Analyze various basic operations of Binary tree and Binary search tree to improve the efficiency.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO10 (K2)	PO11 (K3)	PO12 (K1)
CO1(K3)	3	2	1	1	-	-	-	-	-	-	-	-
CO2(K4)	3	3	2	2	3	-	-	-	-	-	-	-
CO3(K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO4(K4)	3	3	2	2	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1(K3)	3	3	2
CO2(K4)	3	-	3
CO3(K3)	3	-	2
CO4(K4)	3	3	3

List of Experiments:

For the following experiments use C program to implement,

1. Searching

1.1) Recursive and non recursive functions to performLinear search for a Key value in a given list.
1.2) Recursive and non recursive functions to performBinary search for a Key value in a given list.

2. Sorting-I

2.1) Bubble sort, to sort a given list of integers.
2.2) Insertion sort, to sort a given list of integers.
2.3) Selection sort, to sort a given list of integers.
3. Quick sort, to sort a given list of integers.

4. Sorting-II

4.1) Merge sort, to sort a given list of integers.

4.2) Radix sort, to sort a given list of integers.

5. Stack operations using arrays.
6. Stack operations to evaluate the postfix expression.
7. Queue operations using arrays.
8. Singly linked list and its operations.
9. Doubly linked list and its operations.

10. Linked Lists

10.1) Stack operations using Linked List.

10.2) Queue operations using Linked List.

11. Binary tree traversals :inorder, preorder and postorder.
12. Binary Search Tree and its operations.

List of Augmented Experiments:
(Any 2 of the following experiments can be performed)

13. Balanced brackets problem using stack. A bracket is considered to be any one of the following characters: (,), {, }, [, or]. Two brackets are considered to be a matched pair if the an opening bracket (i.e., (, [, or {) occurs to the left of a closing bracket (i.e.,),], or }) of the exact same type.
14. Simplify an algebraic string of characters having '+', '-' operators and parenthesis. You need to print the simplified equation without the parenthesis.
    Example: Input: a-(b+c)
    Output: a-b-c
15. Represent Sparse Matrices using Linked Lists.
16. Towers of Hanoi Problem. Tower of Hanoi is a mathematical puzzle where we have three rods and n disks. The objective of the puzzle is to move the entire stack to another rod, obeying the following simple rules:
    a) Only one disk can be moved at a time.
    b) Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack or on an empty rod.
    c) No larger disk may be placed on top of a smaller disk.

Reference Books:

1. Data Structures And Algorithm Analysis In C, Mark Allen Weiss,2nd Edition.
2. Data Structures AndAlgorithms.A.V.Pai,TMH,2008.
3. Data Structures With C,SeymourLipsehutz, TMH.

Web Links:

1. http://nptel.ac.in/courses/106102064/
2. http://www.udacity.com/
3. http://www.courseera.com/
4. http://www.geeksforgeeks.org/data-structures/
5. http://www.studytonight.com/data-structures/

PYTHON PROGRAMMING LAB
(Common to CSE & IT)

	L	T	P	C
II Semester	**L**	**T**	**P**	**C**
Course Code: 201ES2L14	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To enable the students apply basic concepts in python.

COB 2: To make the students learn the fundamentals of control statements and strings

COB 3: To train students utilize data structures and associated methods, functions in Python.

COB 4: To impart the knowledge on Object Oriented Concepts and file handling.

COB 5: To illustrate the concepts of exceptions and GUI application development.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Develop programs using conditional and iterative statements.

CO 2: Make use of different data structures in solving complex problems.

CO 3: Apply the concepts of functions and files in solving real time applications.

CO 4: Build applications for handling exceptions.

CO 5: Develop GUI applications.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	1	-	3	-	-	3	-	-	3	-
CO2 (K3)	3	2	1	-	3	-	-	3	-	-	-	-
CO3 (K3)	3	2	1	-	3	-	-	3	-	-	3	-
CO4 (K3)	3	3	1	2	3	-	-	3	-	-	-	-
CO5 (K3)	3	2	1	-	3	-	-	3	-	-	3	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K3)	3	3	2
CO2 (K3)	3	2	2
CO3 (K3)	3	3	2
CO4 (K3)	3	-	2
CO5 (K3)	3	-	2

List of Experiments:

1. Basic Programs

   1.1) Write a program that asks the user for a weight in kilograms and converts it to pounds. There are 2.2 pounds in a kilogram.

   1.2) Write a program that asks the user to enter three numbers (use three separate input statements). Create variables called total and average that hold the sum and average of the three numbers and print out the values of total and average.

2. **Control Flow**

   2.1) Write a program that uses a for loop to print the numbers 8, 11, 14, 17, 20, . . . , 83, 86, 89.

   2.2) Write a program that asks the user for their name and how many times to print it. The program should print out the user’s name the specified number of times

3. **Control Flow Continued**

   3.1) Use a for loop to print a triangle like the one below. Allow the user to specify how high the triangle should be.

   ```
   *
   **
   ***
   ****
   ```

   3.2) Generate a random number between 1 and 10. Ask the user to guess the number and print a message based on whether they get it right or not.

   3.3) Write a program that asks the user for two numbers and prints Close if the numbers are within .001 of each other and Not close otherwise.

4. **Strings**

   4.1) Write a program that asks the user to enter a word and prints out whether that word contains any vowels.

   4.2) Write a program that asks the user to enter two strings of the same length. The program should then check to see if the strings are of the same length. If they are not, the program should print an appropriate message and exit. If they are of the same length, the program should alternate the characters of the two strings. For example, if the user enters abcde and ABCDE the program should print out AaBbCcDdEe

   4.3) Write a program that asks the user for a large integer and inserts commas into it according to the standard American convention for commas in large numbers. For instance, if the user enters 1000000, the output should be 1,000,000.

   4.4) In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.

5. **Data structure**

   5.1) Write a program that generates a list of 20 random numbers between 1 and 100.

   (a) Print the list.

   (b) Print the average of the elements in the list.

   (c) Print the largest and smallest values in the list.

   (d) Print the second largest and second smallest entries in the list

   (e) Print how many even numbers are in the list.

   5.2**)** Write a program that asks the user for an integer and creates a list that consists of the factors of that integer.

   5.3)Write a program that generates 100 random integers that are either 0 or 1. Then find the longest run of zeros, the largest number of zeros in a row. For instance, the longest run of zeros in [1,0,1,1,0,0,0,0,1,0,0] is 4.

6. **Data Structure-Continued**

   6.1) Write a program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].

   6.2) Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements,it is much shorter with lists and it is also easier to add new conversions if you use lists

6.3) Write a python script to perform following operations:

i) Create a matrix and print it

ii) Perform Addition of 2 matrices

iii)Perform multiplication of 2 matrices

7. **Functions**

7.1) Write a function called sum_digits that is given an integer num and returns the sum of the digits of num.

7.2) Write a function called first_diff that is given two strings and returns the first location in which the strings differ. If the strings are identical, it should return -1.

7.3) Write a function called number_of_factors that takes an integer and returns how many factors the number has.

7.4) Write a function called is_sorted that is given a list and returns True if the list is sorted and False otherwise.

8. **Functions-Continued**

8.1) Write a function called root that is given a number x and an integer n and returns x1/n . In the function definition, set the default value of n to 2.

8.2) Write a function called primes that is given a number n and returns a list of the first n primes. Let the default value of n be 100.

8.3) Write a function called merge that takes two already sorted lists of possibly different lengths, and merges them into a single sorted list.

(a) Do this using the sort method.

(b) Do this without using the sort method

8.4) Write a program that asks the user for a word and finds all the smaller words that can be made from the letters of that word. The number of occurrences of a letter in a smaller word can’t exceed the number of occurrences of the letter in the user’s word.

9. **Files**

9.1) Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.

9.2) Write a program that reads a list of temperatures from a file called temps.txt, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt.

9.3) Write a program to count frequency of characters in a given file.

10. **OOP**

10.1) Write a class called Product. The class should have fields called name, amount, and price, holding the product’s name, the number of items of that product in stock, and the regular price of the product. There should be a method get_price that receives the number of items to be bought and returns a the cost of buying that many items, where the regular price is charged for orders of less than 10 items, a 10% discount is applied for orders of between 10 and 99 items, and a 20% discount is applied for orders of 100 or more items. There should also be a method called make_purchase that receives the number of items to be bought and decreases amount by that much.

10.2) Write a class called Time whose only field is a time in seconds. It should have a method called convert_to_minutes that returns a string of minutes and seconds formatted as in the following example: if seconds is 230, the method should return '5:50'. It should also have a method called convert_to_hours that returns a string of hours, minutes, and seconds formatted analogously to the previous method.

10.3) Write a class called Converter. The user will pass a length and a unit when declaring an object from the class—for example, c = Converter(9,'inches'). The possible units are inches, feet, yards, miles, kilometers, meters, centimeters, and millimeters. For each of these units there should be a method that returns the length converted into those units. For example, using the Converter object created above, the user could call c.feet() and should get 0.75 as the result.

11. **OOP Continued**

11.1) Write a Python class to implement pow(x, n)

11.2) Write a Python class to reverse a string word by word.

12. **GUI & Exception Handling**

12.1) Write a program that opens a file dialog that allows you to select a text file. The program then displays the contents of the file in a textbox.

12.2) Write a program to demonstrate Try/except/else.

12.3) Write a program to demonstrate try/finally and with/as.

List of AUGMENTED EXPERIMENTS
(Minimum of 2 experiments have to be performed)

13. Write a recursive python function which returns True if the input is well-formatted with respect to the list labels. Else it should return False.

An input is a well-formatted with respect to the labels, if it follows below conditions:

(a) input item is a list

(b) input item has length at least two

(c) input’s first item is in the list labels

(d) each of the remaining items in input is either a string or a well-formatted list

Refer the below table for possible of input items

Sample input		**Expected output**
input item	**list label**	
['VP', ['V', 'eat']]	['VP', 'V']	TRUE
['NP', ['N', 'a', 'or', 'b'], 'c']	['NP', 'V', 'N']	TRUE
[1, [2, 'oui', [1, 'no']], 'no']	[1,2]	TRUE
['VP', ['V', 'eat']]	['VP']	FALSE
['VP', ['V']]	['VP', 'V']	FALSE

14. **Hangman**

The Goal: Despite the name, the actual “hangman” part isn’t necessary. The main goal here is to create a sort of “guess the word” game. The user needs to be able to input letter guesses. A limit should also be set on how many guesses they can use. This means you’ll need a way to grab a word to use for guessing. (This can be grabbed from a pre-made list. No need to get too fancy.) You will also need functions to check if the user has actually inputted a single letter, to check if the inputted letter is in the hidden word (and if it is, how many times it appears), to print letters, and a counter variable to limit guesses.

15. Write a program to find the greatest number that can be formed by using given set of numbers.

16. Write a program to find sum of digits of a number till you get single digit sum.

. Example:
Input : 142 (Hint: 1+4+2=7)
Output : 7
Input : 4683 (Hint: 4+6+8+3=21 =>2+1=3)
Output : 3

17. Write a program to count how many times each word present in a file.

Reference Books:

1. Python for Everybody Exploring Data in Python 3, Charles Russell Severance, SueBlumenberg, 2016
2. Learning Python, Mark Lutz, Orielly, 2013.
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson, 2017

Web Links :

1. https://www.hackerrank.com/
2. https://www.codechef.com/
3. https://www.topcoder.com/
4. https://code-cracker.github.io/ *

****

PROFESSIONAL COMMUNICATION SKILLS LAB

(Common to CE, EEE, ECE, CSE & IT)

II Semester **L T P C**

Course Code: 201MC2L01 **0 0 3 0**

Course Objectives:

COB 1: Tofacilitatecomputer-aided multi-media instructionenablingindividual and independent language learning.

COB 2: To improve the fluency in spoken English and neutralize mother tongue Influence.

COB 3: To train students to use language appropriately.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Illustrateinterpersonalskillsusing languageconfidently andeffectively for personal and profession growth.

CO 2: Make use of effective delivery strategies to select, compile, andsynthesize information for an oral presentation.

CO 3: Demonstrate in mock interviews, mock group discussion and publicspeaking.

CO 4: Identify communicative competency to respond to others in differentsituations.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO2 (K3)	-	-	-	-	-	-	-	-	-	2	-	-
CO3 (K2)	-	-	-	-	-	-	-	-	-	3	-	-
CO4 (K3)	-	-	-	-	-	-	-	-	-	2	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K2)	-	-	-
CO2 (K3)	-	-	-
CO3 (K2)	-	-	-
CO4 (K3)	-	-	-

PRACTICE 1:
Body Language

PRACTICE 2:
Dialogues

PRACTICE 3:
Presentation Skills

PRACTICE 4:
Group Discussion

PRACTICE 5:
Interviews and Telephonic Interviews.

PRACTICE 6:
Debates

Reference Books:

1. Strengthen your Communication Skills by Dr.M.Hari Prasad, Dr.SalivendraJ.Raju and Dr.G.Suvarna Lakshmi,Maruthi Publications.
2. English for Professionals by Prof Eliah,B.S Publications, Hyderabad.
3. A Hand book of English for Professionals by Prof Eliah, B.S Publications.
4. Effective Technical Communication by M. Ashraf Rizvi, Tata Mcraw – Hill Publishing Company.
5. Cornerstone, Developing soft skills,Pearson Education.

Web Links:

1. https://edu.gcfglobal.org/en/business-communication/how-to-deescalate-an-argument-at-work/1/
2. https://www.youtube.com/watch?v=NNamZZsggM4
3. https://www.inc.com/guides/how-to-improve-your-presentation-skills.html
4. https://www.skillsyouneed.com/ips/interview-skills.html
5. https://www.sfu.ca/cmns/130d1/HOWTODEBATE.htm

****

CONSTITUTION OF INDIA

(Common to all branches)

	L	T	P	C
II Semester	**L**	**T**	**P**	**C**
Course Code: 201MC2T02	**2**	**0**	**0**	**0**

Course Objectives:

COB 1: To enable the student to interpret the importance of constitution.

COB 2: To facilitate the students to illustrate the structure of executive, legislature and judiciary.

COB 3: To allow the students to classify philosophy of fundamental rights and duties.

COB 4: To facilitate the students to outline the autonomous nature of constitutional bodies like Supreme Court, High Court, Comptroller and Auditor General of India and Election Commission of India.

COB 5: To enable the student to interpret the central and state relations of finance and administration.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Explain historical background of the constitution making and its importance for building a democratic India.

CO 2: Compare the functioning of three wings of the government i.e.., executive, legislative and judiciary.

CO 3: Interpret the value of the fundamental rights and duties for becoming good citizen of India.

CO 4: Compare the decentralization of power between central, state and local self-government.

CO 5: Extend the knowledge in strengthening of the constitutional institutions like CAG, Election Commission and UPSC for sustaining democracy.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	-	-	-	-	-	-	-	-	2	-	-	-
CO2 (K2)	-	-	-	-	-	-	-	-	3	-	-	-
CO3 (K2)	-	-	-	-	-	-	-	-	3	-	-	-
CO4 (K2)	-	-	-	-	-	-	-	-	3	-	-	-
CO5 (K3)	-	-	-	-	-	-	-	-	2	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K3)	-	-	-
CO2 (K2)	-	-	-
CO3 (K2)	-	-	-
CO4 (K2)	-	-	-
CO5 (K3)	-	-	-

UNIT-I

Introduction to Indian Constitution: Constitution' meaning of the term, Indian Constitution - Sources and constitutional history, Features - Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

UNIT-II

Union Government and its Administration Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha, The Supreme Court and High Court: Powers and Functions.

UNIT-III

State Government and its Administration Governor - Role and Position - CM and Council of ministers, State Secretariat: Organization, Structure and Functions.

UNIT-IV

A .Local Administration - District's Administration Head - Role and Importance, Municipalities - Mayor and role of Elected Representative - CEO of Municipal Corporation PachayatiRaj: Functions PRI: Zila Panchayat, Elected officials and their roles, CEO Zila Panchayat: Block level Organizational Hierarchy - (Different departments), Village level - Role of Elected and Appointed officials - Importance of grass root democracy.

UNIT-V

Election Commission: Election Commission- Role of Chief Election Commissioner and Election Commissionerate State Election Commission, Functions of Commissions for the welfare of SC/ST/OBC and women.

Text Books:

1. Durga Das Basu, Introduction to the Constitution of India, Prentice – Hall of India Pvt. Ltd. NewDelhi.
2. Subash Kashyap, Indian Constitution, National Book Trust.

Reference Books:

1. J.A. Siwach, Dynamics of Indian Government & Politics.
2. D.C. Gupta, Indian Government and Politics.
3. H.M. Sreevai, Constitutional Law of India, 4th edition in 3 volumes (Universal Law Publication).
4. J.C. Johari, Indian Government and Politics Hans.

Web Links:

1. nptel.ac.in/courses/109104074/8
2. nptel.ac.in/courses/109104045/
3. nptel.ac.in/courses/101104065/
4. www.hss.iitb.ac.in/en/lecture-details
5. www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indian-constitution

***

ADVANCED DATA STRUCTURES

(Common to CSE & IT)

		L	T	P	C
III Semester					
Course Code:	**201CS3T01**	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To make the students learn External Sorting and Hashing Techniques.
COB 2: To impart the knowledge on Priority Queues.
COB 3: To provide knowledge on Efficient Binary Search trees and Multiway Search Trees.
COB 4: To enable the students know the significance of Digital Search Trees.
COB 5: To facilitate the students learn String Processing Algorithms.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Demonstrate the External Sorting and Hashing.
CO 2: Illustrate the concepts of Priority Queues.
CO 3: Analyze the Efficient Binary Search trees and Multiway Search Trees.
CO 4: Compare the Digital Search Structures.
CO 5: Apply the String Matching Algorithms to real time applications.

Mapping of course outcomes with program outcomes:

CO/PO	PO1 (K3)	PO2 (K4)	PO3 (K5)	PO4 (K5)	PO5 (K3)	PO6 (K3)	PO7 (K2)	PO8 (K3)	PO9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	3	2	-	-	-	2	-	-	-	-	-	-
CO2 (K2)	3	3	2	2	-	3	-	-	-	-	-	-
CO3 (K4)	2	1	-	-	-	2	-	-	-	-	-	-
CO4 (K2)	3	3	3	3	-	3	-	-	-	-	-	-
CO5 (K3)	2	1	2	2	-	3	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K2)	-	-	3
CO2 (K2)	3	-	3
CO3 (K4)	-	2	3
CO4 (K2)	2	2	-
CO5 (K3)	3	3	-

UNIT-I:

External Sorting: Introduction, K-way Merge Sort, Buffer Handling for parallel Operation, Run Generation, Optimal Merging of Runs, Huffman Tree.

Hashing: Introduction to Static Hashing, Hash Tables, Hash Functions, Different Hash Functions, Collision Resolution Techniques, Dynamic Hashing.

UNIT-II:

Priority Queues (Heaps): Introduction, Binary Heaps-Model and Simple Implementation, Basic Heap Operations, Other Heap Operations, Applications of Priority Queues, Binomial Heaps (or Queues), Binomial Heap Structure and Implementation, Binomial Queue Operations.

UNIT-III:

Efficient Binary Search Trees: Self-balancing Binary Search Tree, AVL Trees, Rotations-LL, RR, LR and RL, Searching, Insertion, Deletion operations on AVL Trees, Red-Black Tree, Properties and Representation of Red-Black Trees, Insertion and deletion operations on Red-Black Trees, Applications of Red-Black Trees.

UNIT-IV:

Multiway Search Trees: M-Way Search Trees Definition and Properties, B-Tree Definition and Properties, Searching, Insertion and Deletion operations on B-Trees, B+ Tree, Insertion and Deletion operations on B+ Trees.

Digital Search Structures: Introduction to Digital Search Tree, Operations on Digital Search Trees- Insertion, Searching, and Deletion.

UNIT-V:

Digital Search Structures: Binary Tries, Compressed Binary Trie, Patricia, Searching Patricia, inserting into Patricia, delete a node from Patricia, Multiway Tries- Definition, Searching a Trie, Compressed Tries, Compressed Tries with Digit Numbers-Searching, Insertion, Deletion.

String Processing: String Operations, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, The Longest Common Subsequence Problem (LCS).

Text Books:

1. Advanced Data Structures, Reema Thareja, S. Rama Sree, Oxford University Press.
2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, Second Edition, Pearson.

Reference Books:

1. Fundamentals Of Data Structures In C, Horowitz, Sahni, Anderson-Freed, Second edition.
2. Data Structures and Algorithms, A. V. Aho, J. E. Hopcroft, and J. D. Ullman, Pearson.
3. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest, Third Edition, The MIT Press.
4. Advanced Data Structures, Peter Brass, Cambridge University Press.

Web Links:

1. https://ocw.mit.edu/courses/...and...data-structures...notes/MIT6_851S12_L1.
2. http://nptel.ac.in/courses/106103069/26
3. https://csd.cs.cmu.edu/course-profiles/15-121-Introduction-to-Data-Structures
4. https://www.cs.purdue.edu/cgvlab/courses/251/lectures/slides/04.03-Pattern atching AndTries.pdf
5. https://www.csie.ntu.edu.tw/~ds/ppt/ch5/chapter5.PPT

OBJECT-ORIENTED PROGRAMMING THROUGH C++

(Common to CSE & IT)

III Semester	**L**	**T**	**P**	**C**
Course Code: 201CS3T02	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To enable the students to learn the basic concepts of C++ programming language.
COB 2: To impart the knowledge on classes, objects, member functions, constructors, destructors compile time and runtime polymorphism in C++.
COB 3: To make the students learn applications of inheritance.
COB 4: To make the student know the importance of Exception handling
COB 5: To demonstrate Generic Programming with Templates.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Compare and Contrast object oriented programming and procedural oriented programming.
CO 2: Summarize the OOPS concepts.
CO 3: Make use of constructor and destructor to initialize and destroy class objects.
CO 4: Apply C++ features such as composition of objects, this pointer, operator overloading, exception handling , compile time and runtime polymorphism
CO 5: Apply inheritance to build real time application
CO 6: Design C++ classes with templates and STL.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K4)	-	3	-	-	1	-	-	-	-	-	-	-
CO2 (K2)	2	1	-	-	2	-	-	-	-	-	-	-
CO3 (K3)	3	-	1	-	-	-	-	-	-	-	-	-
CO4 (K3)	3	2	1	1	-	-	-	-	-	-	-	-
CO5 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO6 (K3)	3	3	3	3	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K4)	-	-	-
CO2 (K2)	2	-	-
CO3 (K3)	2	2	-
CO4 (K3)	2	2	-
CO5 (K3)	2	2	-
CO6 (K3)	3	3	-

UNIT-I:
Introduction to C++: Differences between C and C++, Evolution of C++,the Object Oriented Technology, Disadvantage of Conventional Programming, Key Concepts of Object Oriented Programming, Advantages of OOP, Namespace.

UNIT-II:
Classes and Objects: Declaring Objects- Access Specifiers and their Scope, Defining Member Function, Rules for Inline Functions, static Member Variable, static Member Function, Friend functions, overloading Member Function, Nested class.
Constructors and Destructor Classes in C++: Constructors and Destructors, Introduction to Constructors and Destructor- Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments, Copy Constructor, Destructors, Anonymous Objects.

UNIT-III:
Operator Overloading: Introduction, The Keyword Operator- Overloading Unary Operator, Operator Return Type, Overloading Binary Operator, Overloading using friend function, Overloading Assignment Operator (=),Rules for Overloading Operators.
Inheritance: Introduction, Reusability, Types of Inheritance, Virtual Base Classes, Object as a Class Member, Abstract Classes, Advantages and Disadvantages of Inheritance.
Pointers: Pointer to Class- Pointer Object, this Pointer, Pointer to Derived Classes and Base Class.
Binding Polymorphisms and Virtual Functions: Introduction, Binding in C++, Virtual Functions, Rules for Virtual Function, Abstract classes, Virtual Destructor

UNIT-IV:
Generic Programming with Templates: Introduction, Need for Templates, Definition of class Templates, Normal Function Templates, Overloading of Template Function, Bubble Sort Using Function Templates, Difference between Templates and Macros, Linked list with templates.

UNIT-V:
Exception Handling: Introduction, Principles of Exception Handling, The Keywords try throw and catch, Multiple Catch Statements, Specifying Exceptions.
Overview of Standard Template Library: Introduction, STL Programming Model, Containers, Sequence Containers, Associative Containers, Algorithms, Iterators, Vectors, Lists, Maps.

Text Books:
1. The Complete Reference C++, Herbert Schildt, Fourth Edition, TMH.
2. The C++ Programming Language, Bjarne Stroustr up, Fourth Edition, Pearson.

Reference Books:
1. A First Book Of C++, Gary Bronson, Fourth Edition, Cengage Learning.
2. C++ Primer Plus By Stephen Prata, Sixth Edition, Pearson.
3. C++ Programming:From Problem Analysis To Program Design, Ds Malik, Eighth Edition, Cengage Learning.

Web Links:

1. http://nptel.ac.in/courses/106105151/
2. http://www.geeksforgeeks.org/c-plus-plus/
3. https://www.javatpoint.com/cpp-oops-concepts
4. https://www.tutorialspoint.com/cplusplus/cpp_object_oriented.html

OPERATING SYSTEMS

(Common to CSE & IT)

III Semester	**L T P C**
Course Code: 201CS3T03	**3 0 0 3**

Course Objectives:

COB 1: To impart the knowledge on the Operating System structure, services and System calls.

COB 2: To make the students learn about Process management.

COB 3: To inculcate the knowledge on process synchronization and deadlocks.

COB 4: To illustrate the concept of Memory management.

COB 5: To make the students understand the concepts of Storage management, Protection and security.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Illustrate the basic structure, services, system calls and architectural components of Operating Systems.

CO 2: Analyze various Process Scheduling algorithms and Multi threading models.

CO 3: Demonstrate Inter Process Communication between the processes and deadlocks.

CO 4: Make use of paging, segmentation and virtual memory strategies to allocate memory for the process.

CO 5: Describe the concepts of file system implementation, disk management, Protection and security for system.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO2 (K4)	3	2	1	1	3	-	-	-	-	-	-	-
CO3 (K2)	3	2	1	1	3	-	-	-	-	-	-	-
CO4 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO5 (K2)	3	2	1	1	3	-	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO/PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K2)	3	3	2
CO2 (K4)	3	3	2
CO3 (K2)	3	3	2
CO4 (K3)	3	3	2
CO5 (K2)	3	3	2

UNIT-I:

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, Computing environments, Open-Source Operating Systems. **System Structures:** Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, System Boot.

UNIT-II:

Process Concept: Process scheduling, Operations on processes, Inter-process communication.

Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling, Thread scheduling.

Multithreaded Programming: Multithreading models, Thread libraries, Threading issues.

UNIT-III:

Inter-process Communication: Race conditions, Critical Regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing, Classical IPC Problems - Dining philosophers problem, Readers and writers problem.

Deadlocks: Resources, Conditions for resource deadlocks, Ostrich algorithm, Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention.

UNIT-IV:

Memory-Management Strategies: Introduction, Swapping, Contiguous memory allocation, Paging, Segmentation.

Virtual Memory Management: Introduction, Demand paging, Copy on-write, Page replacement, Frame allocation, Thrashing, Memory-mapped files, Kernel memory allocation.

File Systems: Files, Directories, File system implementation, management and optimization.

UNIT-V:

Secondary-Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.

System Protection: Goals of protection, Principles and domain of protection, Access matrix, Access control, Revocation of access rights.

System Security: Introduction, Program threats, System and network threats.

Case Studies: Linux, Microsoft Windows.

Text Books:

1. OperatingSystemConcepts,AbrahamSilberschatz,PeterBGalvinandGregGagne,9th Edition, John Wiley and Sons Inc.
2. Modern Operating Systems, Tanenbaum A S, 3rdedition, Pearson Education.

Reference Books:

1. Operating Systems A Concept Based Approach, Dhamdhere D M, 3rdedition, Tata McGraw-Hill.
2. Operating Systems -Internals and Design Principles, Stallings W, 6th edition, Pearson Education.
3. Operating Systems, Nutt G, 3rdedition, Pearson Education.

Web Links:

1. http://nptel.ac.in/downloads/106108101/
2. https://www.coursera.org/learn/iot/lecture/MrgxS/lecture-3-1-operating-systems
3. http://www.geeksforgeeks.org/operating-systems/
4. https://in.udacity.com/auth?next=/course/introduction-to-operating-systems--ud923

SOFTWARE ENGINEERING

(Common to CSE & IT)

III Semester	**L**	**T**	**P**	**C**
Course Code: 201CS3T04	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To impart the knowledge on the Software Engineering Principles, Applications and Process models.

COB 2: To help the students to learn the Requirement Engineering Process.

COB 3: To create awareness on the basic activities of software project management.

COB 4: To interpret the various design models with software requirements.

COB 5: To discuss about Coding principles, various testing techniques and Debugging techniques.

COB 6: To teach the basic concepts of software reliability, quality management, Software maintenance and reusability.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Explain the key facts, concepts, principles, and theories of software & Software Engineering.

CO 2: Compare various software development process models with respective to advantages, disadvantages and applicability.

CO 3: Describe the various responsibilities and activities of Software Project Management.

CO 4: Prepare SRS Document for any real time scenario.

CO 5: Apply various Designs, Coding and testing Principles for developing the software products.

CO 6: Discuss the importance of software reliability, quality management, software maintenance and reusability in Software development.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K2)	2	1	-	-	-	-	3	-	-	-	2	1
CO2 (K2)	3	3	-	2	-	-	-	-	-	-	2	2
CO3 (K2)	3	3	-	2	3	-	3	-	-	3	2	2
CO4 (K2)	2	1	2	-	2	-	3	-	-	3	2	2
CO5 (K3)	3	2	1	1	3	-	2	-	-	-	2	3
CO6 (K2)	3	2	-	1	3	-	3	-	-	3	2	1

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K2)	2	2	-
CO2 (K2)	2	2	-
CO3 (K2)	2	2	-
CO4 (K2)	2	2	-
CO5 (K3)	2	3	-
CO6 (K2)	2	2	-

UNIT-I:

Introduction to Software Engineering: The Nature of Software, The Unique Nature of Web Apps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths.

Software Process: Software Process, Process Classification, Phased Development Life Cycle, Software Development Process Models – Waterfall Model, Iterative Waterfall Model, Prototype Model, Incremental Model, Spiral Model, Agile Process Model and RUP process Model.

Case Study: Survey on different process models including.

i. Advantages and Disadvantages of the models.
ii. Applicability of the model.
iii. Projects developed using various models.

UNIT-II:

Software Project Management: Project Management Essentials, What is Project Management, Software Configuration Management, Risk management.

Project Planning and Estimation: Project Planning Activities, Software Metrics and Measurements, Project Size Estimation, Effort Estimation Techniques.

Case Study: Estimate the effort of the software development using Functional Points and COCOMO Model for any one of the real time problem.

UNIT-III:

Requirements Engineering: Software Requirements, Requirements Engineering Process, Requirements Elicitation and Analysis, Requirements Specification, Requirements Validation, Requirements Management.

Case Study: Create a SRS document for any one of the following Software Projects.

1) Course Registration System
2) Students Marks Analyzing System
3) Online Ticket Reservation System
4) Stock Maintenance

UNIT-IV:

Software Design: Software Design Process, Characteristics of Good Software Design, Design Principles, Modular Design, Software Architecture, Design Methodologies,

Implementation: Coding Principles, Coding Process, Code Verification, Code Documentation.

Case Study: Construct the DFD and CFD for any one of the following Software Projects.

1) Airline Reservation System
2) Students Marks Analyzing System
3) ATM System
4) Library Management System

UNIT-V:

Software Testing: Testing Fundamentals, Test Planning, Black-Box Testing, White-Box Testing, Levels of Testing, Usability Testing, Regression Testing, Debugging Approaches.

Software Quality and Reliability: Software Quality factors, Verification & Validation, Software Quality Assurance, The Capability Maturity Model, Software

Reliability.

Case Study: Design the test cases for any one of the following real time scenarios using White Box & Black Box Testing Techniques.

1) E-Commerce application (Flipkart, Amazon)
2) Mobile Application

Text Books:

1. Software Engineering – Concepts and Practices: Ugrasen Suman, Cengage Learning.
2. Software Engineering: A practitioner's approach, Roger S. Pressman, McGrawHill.

Reference Books:

1. Software Engineering, Lan Sommerville, Ninth Edition, Pearson.
2. Software Engineering, A Precise Approach, Pnkaj Jalote, Wiley India.
3. Fundamentals of Software Engineering, Rajib Mall, Prentice Hall India.
4. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill.
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Pankaj Jalote, An integrated approach to Software Engineering, Springer/ Narosa.

Web Links:

1. https://www.tutorialspoint.com/software_engineering/
2. http://nptel.ac.in/courses/106/105/106105182/
3. https://www.coursera.org/learn/software-processes-and-agile-practices
4. http://www.geeksforgeeks.org/software-engineering-gq/
5. https://www.coursera.org/browse/computer-science/software-development

****

DISCRETE MATHEMATICS

(Common to CSE & IT)

	L	T	P	C
III Semester	**L**	**T**	**P**	**C**
Course Code: 201BS3T13	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To introduce the concepts of mathematical logic, number theory, graph theory in order to develop critical thinking to wards problem solving.

COB 2: To relate the concepts of discrete mathematics to various domains of computer science.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Apply the principles of mathematical logic to statement calculus and predicate calculus.
CO 2: Compute Transitive closure, equivalence classes of binary relations.
CO 3: Solve recurrence relations by various methods.
CO 4: Apply the concepts of graph theory to find euler paths, Hamiltonian paths.
CO 5: Apply the concepts of graph theory to trees.

Mapping of Course Outcomes with Program Outcomes:

CO / PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO 1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 2 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 3 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 4 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO 5 (K3)	3	2	-	-	-	-	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO / PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO 1 (K3)	1	1	-
CO 2 (K3)	1	1	-
CO 3 (K3)	1	1	-
CO 4 (K3)	1	1	-
CO 5 (K3)	1	1	-

UNIT I:

Mathematical Logic:

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus.

Predicate Calculus: Predicate Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT II:

Recurrence Relations: Recurrence Relations, Formation of Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots.

UNIT III:

Binary Relations and Properties: Binary relations, Properties, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive, Closure, War shall Algorithm, Equivalence relation, R-Equivalence class, Partial Ordering Relation, Partially ordered sets, Hasse Diagrams.

UNIT IV:

Graph Theory:

Basic Concepts of Graphs, Matrix Representation of Graphs: Adjacency Matrix, Incidence Matrix, Isomorphic Graphs, Paths and Circuits, Euler and Hamilton Graphs, Planar Graphs and Euler's Formula.

UNIT V:

Trees: Trees-Properties, Spanning trees, BFS Algorithm, DFS Algorithm, Minimal Spanning Trees and Kruskal's Algorithm, Graph Colouring, Chromatic Number.

Text Books:

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.
2. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and R. Manohar, Tata McGraw Hill.
3. Mathematical Foundations of Computer Science, S. Santha, E. V. Prasad, Cengage Publishers.

Reference Books:

1. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby, Sharon Cutler Ross, PHI.
2. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel, T.P. Baker, 2nd Edition, Prentice Hall of India.
3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011.

Web Links:

1. https://nptel.ac.in/courses/106/106/106106094/
2. https://www.smartzworld.com/notes/mfcs-notes-pdf-mathematical-foundation-of-computer-science/
3. https://oufastupdates.com/mathematical-foundation-for-computer-science-mfcs-jntu-notes -m-tech-pdf/
4. https://www.youtube.com/watch?v=P4LyRvlSAUY&list=PLrjkTql3jnm_GEUWTjAS38-hgtT3Dzuag

****

OBJECT-ORIENTED PROGRAMMING THROUGH C++ LAB

(Common to CSE & IT)

	L	T	P	C
III Semester	**L**	**T**	**P**	**C**
Course Code: 201CS3L01	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To impart knowledge on classes, object constructor and destructor.
COB 2: To illustrate how to overload functions and operators in C++.
COB 3: To enable the students in developing programming skills using advanced features of C++ such as inline, friend functions, abstraction, encapsulation, polymorphism, this pointer and virtual destructor.
COB 4: To demonstrate exception handling in C++.
COB 5: To implement C++ programs using template and STL in C++.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Use Control Structures and modular programming in solving complex problems.
CO 2: Apply object oriented techniques to solve computing problems.
CO 3: Develop C++ classes for code reuse through inheritance.
CO 4: Apply exception handling technique to handle various errors.
CO 5: Develop C++ programs using Inline, friend functions, Reference variable, this pointer, operator Overloading, static and dynamic binding, template and STL.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	1	1	-	3	-	-	-	-	-	-
CO2 (K3)	2	2	1	1	3	3	-	-	3	-	-	-
CO3 (K3)	3	2	1	-	3	3	-	-	-	-	3	-
CO4 (K3)	3	2	1	1	3	3	-	-	-	-	3	3
CO5 (K3)	3	2	1	1	3	3	3	-	-	-	3	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	3	2	2
CO2 (K3)	3	2	2
CO3 (K3)	3	2	2
CO4 (K3)	3	2	2
CO5 (K3)	3	2	2

List of Experiments:

1. **CONTROL FLOW:**
    1.1) Develop a C++ program to find the roots of a quadratic equation.
    1.2) Develop a C++ program to find factorial of a given number using recursion

2. **VARAIBLE AND SCOPE:**
    2.1) Develop a C++ program to illustrate scope resolution and namespaces.
    2.2) Develop a C++ program illustrating Inline Functions.

3. **CLASS AND OBJECT:**
   - 3.1) Develop a C++ program demonstrating a Bank Account with necessary data members and member functions.
   - 3.2) Develop a C++ program for illustrating Access Specifiers :public and private.
   - 3.3) Develop a C++ program to illustrate this pointer.

4. **FUNCTIONS:**
   - 4.1) Develop a C++ program illustrate function overloading.
   - 4.2) Develop a C++ program to illustrate the use of default arguments.
   - 4.3) Develop a C++ program illustrating friend function.

5. **CONSTRUCTOR AND DESTRUCTOR:**
   - 5.1) Develop a C++ Program to illustrate the use of Constructors and Destructors.
   - 5.2) Develop a C++ program illustrating Constructor overloading.
   - 5.3) Develop a C++ program illustrating Copy Constructor.

6. **OPERATOR OVERLOADING:**
   - 6.1) Develop a C++ program to Overload Unary, and Binary Operators using member function.
   - 6.2) Develop a C++ program to Overload Unary, and Binary Operators using friend function.
   - 6.3) Develop a case study on Overloading Operators and Overloading Functions. (150 Words).

7. **INHERITANCE:**
   - 7.1) Develop C++ Programs to incorporate various forms of Inheritance
     i. Single Inheritance
     ii. Multiple Inheritances
     iii. Multi-level inheritance
     iv. Hierarchical Inheritance
     v. Hybrid inheritance
   - 7.2) Develop a C++ program in C++ to illustrate the order of execution of constructors and destructors in inheritance.

8. **POINTERS:**
   - 8.1) Develop a C++ program to illustrate object as a class member.
   - 8.2) Develop a C++ program to illustrate pointer to a class.
   - 8.3) Develop a C++ program to illustrate Virtual Base Class.

9. **POLYMORPHISM:**
   - 9.1) Develop a C++ program to illustrate virtual functions.
   - 9.2) Develop a C++ program to illustrate runtime polymorphism.
   - 9.3) Develop a C++ program to illustrate pure virtual function and calculate the area of different shapes by using abstract class.

10. **TEMPLATES:**
    - 10.1) Develop a C++ Program illustrating function template.
    - 10.2) Develop a C++ Program illustrating template class.

10.3) Develop a C++ program to illustrate class templates with multiple parameters.

11. **EXCEPTIONS:**

11.1) Develop a C++ program for handling Exceptions.
11.2) Develop a C++ program to illustrate the use of multiple catch statements.

12. **STL:**

12.1) Develop a C++ program to implement List, Vector and its Operations.
12.2) Develop a C++ program to implement Deque and Deque Operations.
12.3) Develop a C++ program to implement Map and Map Operations.

List of Augmented Experiments:
(Any two of the following experiments can be performed)

13. Develop a C++ program for flight booking system
14. Develop Qt application containing slider and spin box in which a slider responds to changes in the spin box.
15. Develop a Qt application for creating a text pad.
16. Develop a C++ program with maximum of 20 characters, that your user will be guessed and will show only asterisks (*) on the screen. The user will input or enter one character at a time. And for every correct character, the asterisk will be replaced by that character until all the characters or the mystery word/s will reveal. Your program will accept a maximum three (3) errors or mistakes in entering/inputting character otherwise the mystery word/s will be viewed.

    Sample Output:

```
Output: *******
Enter your character: e
Output: ***e**e
Enter your character: a Output: sorry! the character is not existing. you still have 2 chances
 Enter your character: s
Output: s**e**e
Enter your character: c
Output: sc*e*ce
Enter your character: i
Output: scie*ce
Enter your character: n
Output: science
```

Reference Books:

1. C++ Primer Plus by Stephen Prata, Sixth Edition, Pearson.
2. C++ GUI Programming with Qt4, Jasmin Blanchette, Mark Summerfield, Second Edition, Prentice Hall Press.
3. C++ for Programmers, Paul J. Deitel, Harvey M. Deitel, Pearson.

Web Links:

1. http://en.cppreference.com/w/cpp/links/libs
2. https://www.daniweb.com/digital-media/ui-ux-design/threads/113591/trying-to-run-ac-program-through-a-web-link
3. http://www.yolinux.com/TUTORIALS/LinuxTutorialC++.html
4. https://github.com/fffaraz/awesome-cpp
5. http://www.techsystemsembedded.com/cpp_links.php

OPERATING SYSEMS LAB
(Common to CSE & IT)

		L	T	P	C
III Semester		**L**	**T**	**P**	**C**
Course Code:	**201CS3L02**	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To enable the students design and apply the process management concepts.
COB 2: To impart the knowledge of System calls and Deadlock handling algorithm.
COB 3: To demonstrate the memory management concepts.
COB 4: To enable the students to use Page replacement and file allocation.
COB 5: To make the students to use Pthreads library.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Simulate various process scheduling algorithms.
CO 2: Experiment with various system calls and deadlock algorithm.
CO 3: Develop algorithms for memory management.
CO 4: Implement Page replacement algorithms and file allocation strategies.
CO 5: Make use of Pthread library for thread concurrent execution.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO2 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO3 (K3)	3	2	1	-	-	-	-	-	-	-	-	-
CO4 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO5 (K3)	3	2	1	-	-	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K3)	3	3	2
CO2 (K3)	3	3	2
CO3 (K3)	3	3	2
CO4 (K3)	3	3	2
CO5 (K3)	3	3	2

List of Experiments:

1. Simulate the following CPU scheduling algorithms:
   (a) FCFS (b) SJF
2. Simulate the following CPU scheduling algorithms:
   (a) Priority (b) Round Robin
3. Multiprogramming-Memory management-Implementation of fork (), wait (), exec() and exit (), System calls
4. Simulate the Multiprogramming with a fixed number of tasks (MFT)
5. Simulate the Multiprogramming with a variable number of tasks (MVT)
6. Simulate Bankers Algorithm for Dead Lock Avoidance
7. Simulate the FIFO page replacement algorithm
8. Simulate the LRU page replacement algorithm
9. Simulate the following File allocation strategies (a) Sequenced (b) Indexed (c)

Linked

10. Write a C program that illustrates two processes communicating using shared memory
11. Write C program to create a thread using pthreads library and let it run its function.
12. Write a C program to illustrate concurrent execution of threads using pthreads library

List of Augmented Experiments:

(Any two of the following experiments can be performed)

13. Simulate Bankers Algorithm for Dead Lock Prevention
14. Simulate Best-Fit contiguous memory allocation technique
15. Simulate FCFS Disk Scheduling algorithm
16. Write a C program to simulate producer and consumer problem using semaphores

Reference Books:

1. Operating System Concepts, Abraham Silberschatz, Peter BGalvin and GregGagne, 9th Edition, John Wiley and Sons Inc..
2. Modern Operating Systems, Tanenbaum A S, 3rdedition, Pearson Education.
3. Operating Systems: A Modern Perspective, Gary J. Nutt.

Web Links:

1. http://nptel.ac.in/downloads/106108101/
2. https://www.coursera.org/learn/iot/lecture/MrgxS/lecture-3-1-operating-systems
3. http://www.geeksforgeeks.org/operating-systems/
4. https://in.udacity.com/auth?next=/course/introduction-to-operating-systems--ud923

***

UNIX AND SHELL PROGRAMMING LAB

(Common to CSE & IT)

	L	T	P	C
III Semester	**L**	**T**	**P**	**C**
Course Code: 201CS3L03	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To create awareness on UNIX environment.
COB 2: To enable the students work with various basic UNIX commands
COB 3: To solve problems using bash for shell scripting
COB 4: To facilitate the students, develop Shell Scripts.
COB 5: To impart the knowledge on concepts of UNIX file access control.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Make use of Unix utilities and perform basic shell control operations on the Unix utilities.
CO 2: Make use of various commands in UNIX to control various resources like file, network, disk etc.
CO 3: Develop Shell Script using Shell commands.
CO 4: Construct AWK Script using AWK commands.
CO 5: Apply system calls for File Management, Process Management and IPC

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO 11 (K3)	PO 12 (K1)
CO1 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO2 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO3 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO4 (K3)	3	2	-	-	-	-	-	-	-	-	-	-
CO5 (K3)	3	2	1	1	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K2)	3	3	-
CO2 (K2)	3	3	-
CO3 (K3)	-	-	-
CO4 (K2)	-	-	-
CO5 (K3)	3	3	-

List of Experiments:

1. a) Study of Unix/Linux general purpose utility command list: man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.
   b) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
   c) Study of Unix/Linux file system (tree structure).
   d) Study of .bashrc, /etc/bashrc and Environment variables.

2. a) Use the cat command to create a file containing the following data.Call it mytable use tabs to separatethe fields.

1425	Ravi	15.65
4320	Ramu	26.27
6830	Sita	36.15
1450	Raju	21.86

b) Study of vi editor

c) Use the cat command to display the file, my table.

d) Use the vi command to correct any errors in the file, my table.

e) Use the sort command to sort the file my table according to the first field. Call the sorted file my table (same name).

f) Print the file my table.

3. a) Write a C program that makes a copy of a file using standard I/O, and system calls.

b) Write a C program to emulate the UNIX ls –l command.

c) Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls –l | sort.

4. **Shell Script**

a) Write a shell script that takes a command –line argument and reports on whether it is directory, a file or something else.

b) Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase provided they exist in the current directory.

5. **Shell Script**

a) Write a shell script that determines the period for which a specified user is working on the system.

b) Write a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.

6. **Shell Script**

Write a shell script that computes the gross salary of a employee according to the following rules:

i) If basic salary is < 1500 then HRA =10% of the basic and DA =90% of the basic.

ii) If basic salary is >=1500 then HRA =Rs500 and DA=98% of the basic. The basic salary is entered interactively through the key board.

7. **Shell Script**

a) Write a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number.

b) Write a shell script which will display Armstrong number from given arguments.

8. **Shell Script**

Write an interactive file-handling shell program. Let it offer the user the choice of copying, removing, renaming, or linking files. Once the user has made a choice, have the program ask the user for the necessary information, such as the file name, new name and so on.

9. a) Write shell script that takes a login name as command – line argument and reports when that person logs in.
   b) Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.

10. Write a C program that takes one or more file or directory names as a command line input and reports the following information on the file:
    i) File type.
    ii) Number of links.
    iii) Read, write and execute permissions.
    iv) Time of last access (Note : Use stat/fstat system calls).
11. Write a C program which supports that child process inherits environment variables, command line arguments, opened' files.
12. Write a shell script to display factorial value from given argument list.

List of Augmented Experiments:

(Any two of the following experiments can be performed)

13. Write a shell script to display reverse number from given argument list.
14. Write a shell script which will display Fibonacci series up to a given number of arguments.
15. Write a shell script to change the ownership of processes.
16. Write a shell script to accept student number, name, marks in 5 subjects. Find total, average and grade. Display the result of the student and store in a file called stu.dat
    Rules: AVG>=80 the grade A
    AVG<80 && AVG>=70 then grade B
    AVG<70 && AVG>=60 then grade C
    AVG<60 && AVG>=50 then grade D
    AVG<50 && AVG>=40 then grade E
    Else grade F

Reference Books:

1. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg. Thomson.
2. Unix Shell Programming, M.G.Venkateshmurthy, Pearson.
3. Your Unix the ultimate guide, Sumitabha Das, 2nd Edition, TMH.
4. The Unix programming Environment, Brain W. Kernighan & Rob Pike, Pearson.
5. Advanced Programming in UNIX Environment, W.Richard Stevens, Stephen, Rago, 3rd Edition.

Web Links:

1. www.iu.hio.no/~mark/unix/unix.html
2. http://www.tutorialspoint.com/unix/
3. www.learnshell.com
4. https://www.informationvine.com

****

APPLICATIONS OF PYTHON-NUMPY

(Skill Oriented Course- I)

(Common to CSE & IT)

		L	T	P	C
III Semester					
Course Code:	**201SO3L05**	**0**	**0**	**4**	**2**

Course Objectives:

COB 1: To provide programming skills in Python package NumPy

COB 2: To provide skills for doing mathematical operations using numpy package

COB 3: To provide skills for doing statistical operations using python

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Explain how data is collected, managed and stored for processing

CO 2: Analyze the workings of various numerical techniques, different descriptive measures of Statistics, correlation and regression to solve the engineering problems.

CO 3: Apply some linear algebra operations to n-dimensional arrays.

CO 4: Perform common data wrangling and computational tasks in Python

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	2	2	1	1	1	-	-	-	-	-	-	2
CO2 (K4)	3	3	3	3	3	-	-	-	-	-	-	2
CO3 (K3)	3	2	2	1	3	-	-	-	-	-	-	2
CO4 (K3)	3	2	2	2	3	-	-	-	-	-	-	2

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1(K3)	PSO2(K3)	PSO3(K4)
CO1(K2)	3	3	3
CO2(K4)	3	2	3
CO3(K3)	3	2	3
CO4(K3)	3	2	3

Perform the following:

1 NumPy Installation using different scientific python distributions (Anaconda, Python(x,y), WinPython, Pyzo)

2 NumPy Basics (np.array, np.arrange, np.linespace, np.zeros, np.ones, np.random.random, np.empty)

3 Arrays (array.shape, len(array), array.ndim, array.dtype, array.astype(type), type(array))

4 Array Manipulation (np.append, np.insert, np.resize, np.delete, np.concatenate, np.vstack, np.hstack)

5 Mathematical Operations(np.add, np.substract, np.divide, np.multiply, np.sqrt, np.sin, np.cos, np.log, np.dot, np.roots) , Statistical Operations(

np.mean, np.median, np.std, array.corrcoef( ))

6 NumPy data types

7 NumPy ndarray

8 NumPy String Operations

9 NumPy Financial functions

10 NumPy Functional Programming

Reference Books:

1. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython by Wes Mc Kinney, O'Reilly Media.
2. Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib, Robert Johansson.

Web Links:

1. https://numpy.org/
2. https://www.geeksforgeeks.org/numpy-linear-algebra/
3. https://www.w3schools.com/python/numpy/default.asp

***

WEB APPLICATION DEVELOPMENT USING FULL STACK FRONTEND DEVELOPMENT – MODULE –I

(Skill Oriented Course- I)

(Common to CSE & IT)

		L	T	P	C
III Semester					
Course Code:	**201SO3L06**	**0**	**0**	**4**	**2**

Course Objectives:

From this Lab, Student will be trained to

COB 1: To illustrate the basic HTML tags and CSS styles to develop basic static web pages.

COB 2: To build web pages with tables and links.

COB 3: To embed audio and video in a web page.

COB 4: To experiment with new tags and styles embedded in HTML5 and CSS3.

COB 5: To build any real time web site using the core concepts of HTML and CSS.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Make use of HTML tags, elements and their attributes for designing static web pages.

CO 2: Experiment with tables to align data and hyperlinks for interactive web pages.

CO 3: Apply form elements for developing Registration and Login webpages.

CO 4: Build a web page by applying appropriate CSS styles to HTML elements.

CO 5: Develop any real time web site using the core concepts of HTML5 and CSS3 along with media.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO 1(K3)	2	3	3	-	3	-	-	-	3	-	-	-
CO 2(K3)	2	3	3	-	3	-	-	-	3	-	-	-
CO 3(K3)	2	3	3	-	3	-	-	-	3	-	-	-
CO 4(K3)	2	3	3	3	3	-	-	-	3	-	-	3
CO 5(K3)	2	3	3	3	3	-	-	-	3	-	-	3

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1(K3)	PSO2(K3)	PSO3(K4)
CO 1(K3)	2	-	-
CO 2(K3)	2	-	-
CO 3(K3)	2	-	-
CO 4(K3)	2	3	-
CO 5(K3)	2	3	-

Perform Experiments Related to The Following Concepts:

HTML: Introduction to HTML, Browsers and HTML, Editor's Offline and Online, Tags, Attribute and Elements, DOCTYPE Element, Comments, Headings, Paragraphs, and Formatting Text, Lists and Links, Images and Tables, Forms

CSS: Introduction CSS, Applying CSS to HTML, Selectors, Properties and Values, CSS Colors and Backgrounds, CSS Box Model, CSS Margins, Padding, and Borders, CSS Text and Font Properties, CSS General Topics.

List of Experiments:

Experiment 1: **Introduction to HTML, Browsers and HTML, Editor's Offline and Online, Tags, Attribute and Elements, DOCTYPE Element**

a. Explain the procedure to create a HTML program using different editors.
b. Explain the procedure to run a HTML program using Browser.
c. List some of the offline and online editors to create and run HTML pages.
d. Explain with syntax, the use of DOCTYPE in HTML page.
e. Write a HTML program to experience the HTML Page Structure.

Experiment 2: **Comments, Headings, Paragraphs. Formatting Text**

a. Write a HTML program, that makes use of basic tags like <html> ,<head>, <title>, <body>, <p>, <hr>, <br>, <h1> to <h6>, <! -- --> tags and their attributes.
b. Write a HTML program, that makes use of text formatting tags like <b>, <i>, <u>, <strong>, <sub>, <super>, <tt>, <pre>.
c. Write a HTML program, to explain the working of <font> tag and its attributes.

Experiment 3: **Lists and Links**

a. Write a HTML program, to explain the working of lists.
   Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.
   Note: Use text to link →https://www.aec.edu.in/
   Use image to link →https://www.aec.edu.in/?p=Gallery

Experiment 4: **HTML Images**

a. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.
b. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique.

Experiment 5: **HTML Tables**

a. Write a HTML program, to explain the working of tables. (use tags

:<table>, <tr>, <th>, <td> and attributes : border, rowspan, colspan)

b. Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).

Experiment 6: Frames and Forms

a. Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame → image, second frame → paragraph, third frame → hyperlink. And also make sure of using “no frame” attribute such that frames to be fixed).

b. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).

Experiment 7: HTML 5

a. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, <div>, <span> tags.

b. Write a HTML program, to embed audio and video into HTML web page.

Experiment 8: Cascading Style Sheets, types of CSS, Selector forms

Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).

Experiment 9: Write a program to apply different types of selector forms

i. Simple selector (element, id, class, group, universal)

ii. Combinator selector (descendant, child, adjacentsibling, general sibling)

iii. Pseudo-class selector

iv. Pseudo-element selector

v. Attribute selector

Experiment 10: Color, Background and CSS Box Model

a. Write a program to demonstrate the various ways you can reference a color in CSS.

b. Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.

Experiment 11: a. Write a program, to explain the importance of CSS Box model usimg

i. Content ii. Border iii. Margin iv. padding

b. Write a program using the following terms related to CSS font and text:

i. font-size ii. font-weight iii. font-style

iv. text-decoration v. text-transformation vi. text-alignment

Experiment 12: a. Write a CSS program, to apply 2D transformations in a web page.
b. Write a CSS program, to apply 3D transformations in a web page.
c. Write a CSS program, to apply Animations in a web page.

List of Augmented Experiments:

(Any two of the following experiments can be performed)

1. Design a web page with all the features of HTML elements.
2. Design a web page with new features of HTML5.
3. Design a web page with all the features of HTML elements and apply CSS styles.
4. Design a web page with new features of HTML5 and CSS3.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson.
2. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, Dream Tech.
3. An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sanda S Katila, Cengage Learning.

Reference Books:

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css

BIOLOGY FOR ENGINEERS

III Semester **L T P C**
Course Code: 201MC3T03 **2 0 0 0**

Course Objectives:

COB 1: To introduce students to modern biology with an emphasis on evolution of biology as a multi-disciplinary field.
COB 2: To introduce students to modern biology with an emphasis on evolution of biology as a multi-disciplinary field.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Apply biological engineering principles, procedures needed to solve real-world problems.
CO 2: Demonstrate the fundamentals of living things, their classification, cell structure and biochemical constituents.
CO 3: Apply the concept of plant, animal and microbial systems and growth in real life Situations.
CO 4: Explain genetics and the immune system to know the cause, symptoms, diagnosis and treatment of common diseases.
CO 5: Demonstrate basic knowledge of the applications of biological systems in relevant industries.

Mapping of course outcomes with program outcomes:

CO / PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO 1 (K3)	-	-	-	-	-	-	2	1	2	2	1	3
CO 2 (K2)	-	-	-	-	-	-	1	2	1	1	2	2
CO 3 (K3)	-	-	-	-	-	-	2	1	2	2	1	3
CO 4 (K2)	-	-	-	-	-	-	1	2	1	1	2	2
CO 5 (K2)	-	-	-	-	-	-	1	2	1	1	2	2

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO1(K3)	PSO2 (K3)	PSO3 (K4)
CO 1 (K3)	1	1	2
CO 2 (K2)	2	2	3
CO 3 (K3)	1	1	2
CO 4 (K2)	2	2	3
CO 5 (K2)	2	2	3

UNIT-I:

Introduction to Life: Characteristics of living organisms-Basic classification-cell theory-structure of prokaryotic and eukaryotic cell-Introduction to biomolecules: definition-general classification and important functions of carbohydrates-lipids-proteins-nucleic acids vitamins and enzymes-genes and chromosome.

UNIT-II:

Biodiversity: Plant System: basic concepts of plant growth-nutrition-photosynthesis and nitrogen fixation-Animal System: elementary study of digestive-respiratory-circulatory-excretory systems and their functions Microbial System: history-types of microbes-economic importance and control of microbes.

UNIT-III:

Genetics And Immune System: Evolution: theories of evolution-Mendel's cell division–mitosis and meiosis-evidence of e laws of inheritance-variation and speciation-nucleic acids as a genetic material- central dogma immunity antigens-antibody-immune response.

UNIT-IV:

Human diseases: Definition- causes, symptoms, diagnosis, treatment and prevention of diabetes, cancer,hypertension, influenza, AIDS and Hepatitis.

UNIT-V:

Biology and its industrial application: Transgenic plants and animals-stem cell, and tissue engineering-bioreactors- biopharming-recombinant vaccines-cloning-drug discovery-biological neural networks-bioremediation-biofertilizer-biocontrolbiofilters-biosensors-biopolymers-bioenergy- biomaterials-biochips-basic biomedical instrumentation.

Text Books:

1. A Text book of Biotechnology, R.C.Dubey, S. Chand Higher Academic Publications, 2013.
2. Diseases of the Human Body, Carol D. Tamparo and Marcia A. Lewis, F.A. Davis Company, 2011.
3. Biomedical instrumentation, Technology and applications, R. Khandpur, McGraw Hill Professional, 2004.

Reference Books:

1. Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011.
2. Cell Biology and Genetics (Biology: The unity and diversity of life Volume I), Cecie Starr, Ralph Taggart, Christine Evers and Lisa Starr, Cengage Learning, 2008.
3. Biotechnology Expanding horizon, B.D. Singh, Kalyani Publishers, 2012.

Web Links:

1. https://www.biology-online.org/
2. http://www.biologyreference.com/
3. http://www.allexperts.com/browse.cgi?catLvl=2&catID=229
4. https://www.library.qmul.ac.uk/subject-guides/biological-sciences

****

FORMAL LANGUAGES AND AUTOMATA THEORY

(Common to CSE & IT)

IV Semester	**L**	**T**	**P**	**C**
Course Code:	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To create awareness on the need and importance of automata theory.
COB 2: To impart knowledge on regular expressions.
COB 3: To discuss the concept of grammar.
COB 4: To demonstrate the working and design of various kinds of automaton.
COB 5: To illustrate the concepts of decidability and undecidability.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Explain the basic concepts of automaton and its properties.
CO 2: Apply interconversion of regular expression and finite automata.
CO 3: Design grammars to produce strings from a specific language.
CO 4: Construct automaton for a given problem.
CO 5: Analyse decidability and undecidability problems.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	3	3	2	2	1	-	-	-	2	-	-	-
CO2 (K3)	3	3	2	2	-	-	-	-	2	-	-	-
CO3 (K3)	3	3	2	2	2	-	-	-	2	-	-	-
CO4 (K3)	3	3	2	2	2	-	-	-	2	-	-	-
CO5 (K4)	3	3	2	3	-	-	-	-	2	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K3)
CO1 (K2)	2	-	-
CO2 (K3)	2	-	-
CO3 (K3)	2	-	-
CO4 (K3)	2	-	-
CO5 (K4)	2	-	-

UNIT-I:

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automation, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT-II:

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, Inter Conversion.

UNIT-III:

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, Є-Productions and Unit Productions, Normal Forms-Chomsky Normal form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.

UNIT-IV:

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

UNIT-V:

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TMs, Church’s Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post’s Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

Text Books:

1. Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008.

Reference Books:

1. Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI.
2. Elements of Theory of Computation, Lewis H.P. & Papadimitriou C.H., Pearson /PHI
3. Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill.
4. Theory of Computation, V. Kulkarni, Oxford University Press.

Web Links:

1. http://nptel.ac.in/courses/111103016/ /
2. https://nptel.ac.in/courses/106/104/106104148/
3. https://www.iitg.ac.in/dgoswami/Flat-Notes.pdf
4. https://www.ics.uci.edu/~goodrich/teach/cs162/notes/
5. https://www.geeksforgeeks.org/introduction-of-finite-automata/

DATABASE MANAGEMENT SYSTEMS

(Common to CSE & IT)

	L	T	P	C
IV Semester	**L**	**T**	**P**	**C**
Course Code:	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To make students gain knowledge on the basic concepts of database systems and its architectures.

COB 2: To enable the students learn various data models.

COB 3: To illustrate the mechanisms involved in normalization of relational databases.

COB 4: To make use of SQL statements for performing operations on databases.

COB 5: To impart knowledge on transaction management and concurrency controls.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Summarize the database characteristics and identify various database architectures.

CO 2: Interpret relational database using SQL.

CO 3: Examine issues in data storage and query processing for appropriate solutions.

CO 4: Make use of normalization techniques for database design.

CO 5: Illustrate the mechanisms of transaction management.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K2)	2	1	-	-	-	-	-	-	-	-	-	-
CO2 (K3)	3	2	1	1	3	-	3	-	-	-	-	-
CO3 (K3)	3	2	-	-	-	-	3	-	-	-	-	-
CO4 (K3)	3	2	1	1	3	-	3	-	-	-	-	-
CO5 (K2)	2	1	1	1	2	-	2	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO1 (K3)	PSO2 (K3)	PSO3 (K4)
CO1 (K2)	2	2	1
CO2 (K3)	-	3	2
CO3 (K3)	3	3	2
CO4 (K3)	3	3	2
CO5 (K2)	-	2	3

UNIT-I:

Introduction: Database system, Characteristics (Database Vs File System), Database Users(Actors on Scene, Workers behind the scene), Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

UNIT-II:

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).

UNIT-III:

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

SQL: Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.

UNIT-IV:

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF), Fifth Normal Form (5NF).

UNIT-V:

Transaction Concept: Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Indexing Techniques: B+ Trees: Search, Insert, Delete algorithms, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing: Tree base Indexing, Comparison of File Organizations, Indexes and Performance Tuning

Text Books:

1. Database Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, TMH.
2. Database System Concepts,5/e, Silberschatz, Korth, TMH.

Reference Books:

1. Introduction to Database Systems, 8/e C J Date, PEA.
2. Database Management System, 6/e Ramez Elmasri, Shamkant B. Navathe, PEA
3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

Web Links:

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://www.geeksforgeeks.org/introduction-to-nosql/
3. https://www.youtube.com/watch?v=wkOD6mbXc2M
4. https://beginnersbook.com/2015/05/normalization-in-dbms/

JAVA PROGRAMMING
(Common to CSE & IT)

IV Semester	**L**	**T**	**P**	**C**
Course Code:	**3**	**0**	**0**	**3**

Course Objectives:

COB 1: To make students understand object oriented programming concepts, and apply them in solving problems

COB 2: To provide knowledge on classes, inheritance, interfaces and packages.

COB 3: To facilitate students in handling exceptions and multithreading.

COB 4: To make the students work with JDBC in developing applications.

Course Outcomes:

At the end of this course the student will be able to:

CO 1: Apply object oriented programming features and concepts for solving given problem.

CO 2: Solve real time problems using the concepts of class, inheritance, interface and packages.

CO 3: Test for runtime exceptions arise in java applications.

CO 4: Develop real time applications using multithreading.

CO 5: Build java applications that interact with database for performing data related operations.

Mapping of Course Outcomes with Program Outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	2	2	2	-	1	-	-	-	-	-	-	-
CO2 (K3)	2	2	3	-	2	-	-	-	-	-	-	-
CO3 (K4)	2	2	3	-	3	-	-	-	-	-	-	-
CO4 (K3)	2	2	3	2	3	-	-	-	-	-	-	-
CO5 (K3)	2	2	3	2	3	-	-	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K4)
CO1 (K3)	-	-	1
CO2 (K3)	-	-	2
CO3 (K4)	-	-	-
CO4 (K3)	3	3	-
CO5 (K3)	3	3	2

UNIT I:

Introduction to Java: History of Java, Java Features, Program Structure, Command Line Arguments, User Input to Programs.

Building Blocks of Java: Identifiers, Data types, Literal Constants, Variables and its Scope, Formatted Output with printf() Method, Operators, Precedence and Associativity of Operators, Type Casting.

Control Statements: Selection Statements**:** if-else, switch, Iteration Statements: while, do-while, for, for each, Transfer Statements: Break, Continue.

UNIT II:

Arrays: Introduction, Declaration and Initialization of Arrays, Accessing Elements of Array. Operations on Array Elements, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Multi-dimensional arrays.

Classes, Objects and Methods: Class Declaration, Creating Objects, Assigning One Object to Another, Methods, Constructors, this keyword, static keyword, final keyword, garbage collector, Access Control, Method Overloading, Constructor Overloading, Parameter Passing, Nested Classes.

String Handling: String Class, Methods for Extracting Characters from Strings, Methods for Comparison of Strings, Methods for Searching Strings, Methods for Modifying Strings, String Buffer Class and its methods, Class String Builder.

UNIT III:

Inheritance: Inheritance, Types of Inheritance, Constructor Method and Inheritance, Super keyword, Method Overriding, Dynamic Method Dispatch, Inhibiting Inheritance of Class Using Final, Abstract Classes.

Interfaces- Defining an interface, Implementing interfaces through classes, Multiple inheritance through interfaces.

UNIT IV:

Packages: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Specifiers, java. lang package, Wrapper Classes.

Exception Handling: Introduction, Importance of try, catch, throw, throws and finally block, Multiple Catch Clauses, Rethrowing Exception, Nested try and catch Blocks, Unchecked Exceptions, Checked Exceptions, Custom Exceptions.

UNIT V:

Multithreading: Introduction, Thread Life Cycle, Creation of Threads, Thread Priorities, Thread Synchronization, Inter-thread Communication- Suspending, Resuming, and Stopping of Threads.

Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, Result Set Interface, Creating JDBC Application.

Text Books:

1. The Complete Reference Java, Herbert Schildt, 8th Edition, TMH.
2. Java one step ahead, Anita seth, B.L.Juneja, First Edition, Oxford.

Reference Books:

1. Introduction to java programming, by Y Daniel Liang, Seventh Edition, Pearson.
2. Core Java: An Integrated Approach, R.Nageswara Rao, Dream tech press.
3. Thinking in Java – Bruce Eckel, Fourth Edition, Prentice Hall.

Web Links:

1. https://nptel.ac.in/courses/106/105/106105191/
2. http://java.sun.com/docs/books/tutorial/
3. http://www.tutorialspoint.com/java
4. http://www.javatpoint.com
5. http://www.w3schools.com/java

DATABASE MANAGEMENT SYSTEMS LAB
(Common to CSE & IT)

IV Semester	**L**	**T**	**P**	**C**
Course Code:	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To impart the fundamentals on the relational data model.
COB 2: To make the students to implement SQL and procedural interfaces to SQL.
COB 3: To enable the students to build three level architectures.
COB 4: To illustrate the components of SQL and its applications.
COB 5: To interpret the concepts and techniques relating to query processing by SQL.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Make use of the concepts of relational model techniques for database design.
CO 2: Construct a database schema for a given problem-domain.
CO 3: Apply Normalization techniques on a database to avoid anomalies.
CO 4: Build queries on a database using SQL DDL/DML commands.
CO 5: Develop PL/SQL stored procedures, stored functions, cursors and packages.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	1	1	3	-	3	-	-	-	-	-
CO2 (K3)	-	2	1	1	3	-	3	-	-	-	-	-
CO3 (K3)	3	2	1	1	3	-	3	-	-	-	-	-
CO4 (K3)	3	2	1	1	3	-	3	-	-	-	-	-
CO5 (K3)	3	2	1	1	3	-	3	-	-	-	-	-

Mapping of Course Outcomes with Program Specific Outcomes:

CO / PSO	PSO 1(K3)	PSO 2(K3)	PSO 3(K2)
CO1 (K3)	3	3	2
CO2 (K3)	-	3	2
CO3 (K3)	3	3	2
CO4 (K3)	3	3	2
CO5 (K3)	3	3	2

List of Experiments:

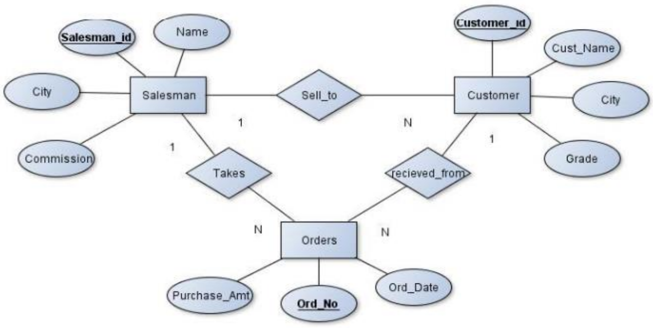
1. Queries for Creating, Altering and Dropping Tables, Views and Constraints.
2. Queries to Retrieve and Change Data: Select, Insert, Delete and Update.
3. 3.1) Queries to facilitate acquaintance of Built-in Functions: String Functions, Numeric Functions, Date Functions and Conversion Functions.
   3.2) Queries using operators in SQL.
4. 4.1) Queries using Group By, Order By, and Having Clauses.
   4.2) Queries on Controlling Data: Commit, Rollback, and Save point.
5. Queries on Joins and Correlated Sub-queries.
6. Queries on Working with Index, Sequence, Synonyms.

7. Queries to Build Views.
8. Write a PL/SQL Code using Basic Variables and Usage of Assignment Operation.
9. Write a PL/SQL Code to Bind and Substitute variables in PL/SQL.
10. Write a PL/SQL block using SQL and Control Structures.
11. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types.
12. Write a PL/SQL Code using Procedures, Functions, Packages.

List of Augmented Experiments:

(Any two of the following experiments can be performed)

13. For a Sales Order Database System, based on the given E-R diagram.
    a) Design a schema by applying functional dependencies.
    b) Apply constraints and verify them.



14. Based on the following schema for a Library Database:
    BOOK (Book_id, Title, Publisher_Name, Pub_Year)
    BOOK_AUTHORS (Book_id, Author_Name)
    PUBLISHER (Name, Address, Phone)
    BOOK_COPIES (Book_id, Branch_id, No-of_Copies)
    BOOK_LENDING (Book_id, Branch_id, Card_No, Date_Out,Due_Date)
    LIBRARY_BRANCH (Branch_id, Branch_Name, Address)
    a. Draw the E-R diagram and show the necessary multiplicity and associations among them.
    b. Draw the Schema diagram and show the necessary associations among them.
15. For a Faculty Database
    EMPLOYEE (EMPID, FName, L Name, Address, Sex, Salary, Dept No)
    DEPARTMENT (Dept No, D Name, HOD_EMPID)
    PROJECT (Proj No, P Name, Dept No)
    WORKS_ON (EMPID, Proj No, Hours)
    Write SQL queries to
    a. To Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
    b. Find the sum of the salaries of all employees of the 'IT' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
16. For a Movie Database:
    ACTOR (Act_id, Act_Name, Act_Gender)
    DIRECTOR (Dir_id, Dir_Name)
    MOVIES (Mov_id, Mov_Title, Mov_Year, Dir_id)
    MOVIE_CAST (Act_id, Mov_id, Role)

RATING (Mov_id, Rev_Stars)

Write SQL queries to

1. List the titles of all movies directed by ‘STEVEN SPIELBERG’.
2. Find the movie names where one or more actors acted in two or more movies.
3. List all actors who acted in a movie before 2015 and also in a movie after 2015 (use JOIN operation).
4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.

Reference Books:

1. SQL, PL/SQL The programming language of ORACLE, Ivan Bayross, Fourth edition, BPB Publication.
2. SQL/PLSQL for ORACLE 9i, P.S.Deshpande, Dreamtech Press.
3. Teach yourself PL/SQL in 21 days, Tom Luers, Timothy Atwood and Jonatham Gennick, First Edition, Techmedia.

Web Links:

1. http://nptel.ac.in/courses/106106093/6
2. http://www.tutorialspoint.com/plsql/
3. https://www.plsql.co/
4. https://www.w3schools.com/sql/
5. http://www.oracle.com/technetwork/database/features/plsql/index.html

***

JAVA PROGRAMMINGLAB
(Common to CSE & IT)

IV Semester	**L**	**T**	**P**	**C**
Course Code:	**0**	**0**	**3**	**1.5**

Course Objectives:

COB1: To impart knowledge on concepts of Object-Oriented Programming.
COB2: To enable students to implement classes, in heritance, interface and package concepts.
COB3: To demonstrate exception handling and multithreading.
COB4: To make the students develop applications using JDBC.

Course Outcomes:

At the end of the Course, Student will be able to:

CO1: Make use of class, inheritance, interface and packages to develop solutions for complex problems.
CO2: Develop error-handling techniques using exception handling.
CO3: Build java applications using Threads.
CO4: Build applications using JDBC connectivity.

Mapping of course out comes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	1	1	3	-	-	-	-	-	-	1
CO2 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO3 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO4 (K3)	3	2	-	1	3	-	-	-	-	-	-	-

Mapping of course out comes with program Specific Outcomes:

CO /PSO	PSO1 (K3)	PSO2 (K3)	PSO3(K4)
CO1 (K3)	-	-	2
CO2 (K3)	-	-	2
CO3 (K3)	-	-	-
CO4 (K3)	3	-	-

List of Experiments:

1)Basic Programs

1.1) Write a Java program to display default value of all primitive data type of JAVA

1.2) Write a Java program to find the discriminant value D and find out the roots of the quadratic equation of theformax^2+bx+c=0.

1.3) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

2)Control Flow Statements

2.1) Write a Java program to select all the prime numbers within the range of 1to100.

2.2) WriteaJavaprogramtoFindthesumofalleventermsintheFibonaccisequenceup To the given range N.

2.3) Write a Java program to check whetherra given number is Armstrong or not.

3)Arrays

3.1) Write a Java program to implement binary search.

3.2) Write a Java program to sort for an element in a given list of elements using bubble sort.

3.3) Write a Java program to sort for an element in a given list of elements using merge sort.

4)Class Mechanism

4.1) Write a Java program to display the details of a person. Personal details should be
given in one method and the qualification details in another method.

4.2) Write a Java program to implement constructor and constructor overloadin

4.3) Write a Java program to implement method overloading.

5)Strings

5.1) Write a Java program to sort given set of strings.

5.2) Write a Java program for using String Buffer to remove or delete a character.

6)Inheritance

6.1) Write a Java program to implement Single Inheritance.

6.2) Write a Java program to implement multi level Inheritance.

6.3) Write a Java program to find the areas of different shapes using abstract classes.

7)Inheritance-continued

7.1) Write a Java program for “super” keyword.

7.2) Take the details of internal exam marks in one Interface. Take the details of external exam marks in another interface. Write a Java program to find the total marks obtained in each subject by a student. (Note: Make use of Multiple Inheritance using interfaces.)

8)Runtime Polymorphism

8.1) Write a JAVA program that implements Runtime polymorphism

9)Packages

9.1) Write a Java program that import and use user defined package.

9.2) Write a Java program to illustrate the use of protected members in a package.

10)Exception Handling

10.1) WriteaJavaprogramtoillustrateexceptionhandlingmechanismusingmultiple catchclauses

10.2) WriteaJavaprogramtomakeuseofBuilt-inanduser-definedExceptionsinhandling a run time exception.

11)Multithreading

11.1) Write a Java program to demonstrate the use of demon thread.

11.2) WriteaJavaprogramthatcreatesthreadsbyextendingThreadclass.Firstthreaddisplay“ Good Morning “every1sec, the second thread displays “Hello“ every2 seconds and the third display “Welcome” every 3 seconds, (Repeat the same by implementing Runnable).

11.3) WriteaJavaprogramtosolveProducer-Consumerproblemusingsynchronization.

12)JDBC

12.1) Write a JDBC program to perform the following operations by connecting to MYSQL database.

I) Inserting Data into Table
II) Updating Data in the Table.
V) Deleting Data From the Table based on a column value.

List of Augmented Experiments:
(Any two of the following experiments can be performed)

13) Create an interface which consists of methods called no of watts consumable, luminescent value, efficiency in percentage. Write classes for different categories of bulbs like LED, tube light and find out which light is efficient in terms of consumption.

14) Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the but tons in selected color.

15) Write a Java program to display analog clock using Applet.

16) Develop an apple that receives an integer in one text field, and compute sits factorial Value and returns it in an other text field, when the button named “Compute” is clicked.

Reference Books:

1. Java How to Program, H.M.Dietel and P.J.Dietel, Pearson Education/PHI, Sixth Edition 2007.
2. Core Java: An Integrated Approach–R.Nageswara Rao, First Edition, JohnWiley andSonsInc.,2015.
3. Java Tutorial: A Short Note on Basics-Sharon Biocca Zakhour, Soumya Kannan, Raymond Gallardo– Fifth Edition, Oracle Corp,2012.
4. Object Oriented Programming using Java–Simon Kendal, First Edition, 2009.
5. Java: The fundamentals of Objects and Classes–David Etheridge, First Edition, 2009.

Web Links:

1. http://www.programmingtutorials.com/java.aspx
2. http://www.javacodegeeks.com
3. http://java.sun.com/developer/onlineTraining/
4. http://java.sun.com/learning
5. http://www.kodejava.org

R PROGRAMMING LAB
(Common to CSE & IT)

	L	T	P	C
IV Semester	**L**	**T**	**P**	**C**
Course Code:	**0**	**0**	**3**	**1.5**

Course Objectives:

COB 1: To enable the students to learn statistical programming, computation, graphics, and modelling.
COB 2: To make the students to write functions and use R in an efficient way.
COB 3: To impart the knowledge about basic types of statistical models.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Make use of online resources for R and import new function packages into the R workspace.
CO 2: Import, review, manipulate and summarize data-sets in R.
CO 3: Explore data-sets to create testable hypotheses and identify appropriate statistical tests.
CO 4: Apply appropriate statistical tests using R.
CO 5: Design and edit visualizations with R.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO2 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO3 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO4 (K3)	3	2	1	1	3	-	-	-	-	-	-	-
CO5 (K3)	3	2	1	1	3	-	-	-	-	-	-	-

Mapping of course outcomes with program Specific Outcomes:

CO / PSO	PSO 1 (K3)	PSO 2 (K3)	PSO 3 (K4)
CO1 (K3)	3	3	-
CO2 (K3)	3	3	-
CO3 (K3)	3	3	-
CO4 (K3)	3	3	-
CO5 (K3)	3	3	-

List of Experiments:

1)Built in functions

1.1) Calculate the cumulative sum(“running total”)of the numbers 2,3,4,5,6
1.2) Print the 1 to 10 numbers in reverse order.

2) Basic Programs

2.1) Write a R program to take input from the user (name and age) and display the values. Also print the version of R installation.
2.2) Write a R program to get the details of the objects in memory.
2.3) Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.

3)Graphics

Write a R program to create a simple bar plot of five subjects marks.

4)Vectors

4.1) Write a R program to get the unique elements of a given string and unique numbers of vector.

4.2) Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3×3 matrix where each column represents a vector. Print the content of the matrix.

4.3) Write a R program to create a matrix from a list of given vectors.

5) Vectors-continued

5.1) Write a R program to append value to a given empty vector.

5.2) Write a R program to multiply two vectors of integers type and length 3.

5.3) Write a R program to find Sum, Mean and Product of a Vector, ignore element like A or NaN.

6)Matrices

6.1) Write a R program to create a 5 x 4 matrix , 3 x 3 matrix with labels and fill the matrix by rows and 2 × 2 matrix with labels and fill the matrix by columns.

6.2) Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.

6.3) Write a R program to find row and column index of maximum and minimum value in a given matrix

7)Arrays

7.1) Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.

7.2)Write a R program to create an array using four given columns, three given rows, and two given tables and display the content of the array.

8)Data frame-I

8.1) Write a R program to create an empty data frame.

8.2) Write a R program to create a data frame from four given vectors.

9) Data frame-II

9.1) Write a R program to create a data frame using two given vectors and display the duplicated elements and unique rows of the said data frame.

9.2) Write a R program to save the information of a data frame in a file and display the information of the file.

10)Lists

10.1) Write a R program to create a list containing a vector, a matrix and a list and give names to the elements in the list. Access the first and second element of the list.

10.2) Write a R program to create a list containing a vector, a matrix and a list and remove the second element.

10.3) Write a R program to select second element of a given nested list.

11)Lists-continued

11.1) Write a R program to merge two given lists into one list.

11.2) Write a R program to create a list named s containing sequence of 15 capital

letters, starting from ‘E’.
11.3) Write a R program to assign new names "a", "b" and "c" to the elements of a given list.

12)Factors
12.1) Write a R program to find the levels of factor of a given vector.
12.2) Write a R program to create an ordered factor from data consisting of the names of months.
12.3) Write a R program to concatenate two given factor in a single factor.

List of Augmented Experiments:
(Any 2 of the following experiments can be performed)

13)The number below are the first ten days of rain fall amounts in 1996.
Read them into a vector using c() function 0.1,0.6,33.8,1.9,9.6,4.3,33.7,0.3,0.0,0.1
Inspect the data and answer the following questions:

a. what was the mean rainfall, how about the standard deviation?
b. calculate the cumulative rainfall(“running total”)over these ten days. confirm that the last value of the vector that this produces is equal to the total sum of the rainfall.
c. which day saw the highest rainfall? hintwhich.max()

14)The weights of five people are given before and after a diet programme are given in the table.

fore					5
ter					

Read the Before and after values into two different vectors called before and after .use R to evaluate the amount of weight lost for each participant. what is the average amount of weight lost?

15)consider A=matrix(c(2,0,1,3),ncol-2)and B=matrix(c(5,2,4,-1),ncol=2)

a.find A+B
b.find A-B

16)Consider a vector 1:K,where K is a positive integer. Write an R command that determines how many elements in the vector are exactly divisible by 3.

Reference Books:

1. Probability And Statistics For Engineering And Sciences, Jay L. Devore, Eighth Edition, Cengage Learning.
2. R Cookbook, Paul Teetor, Oreilly.
3. R In Action, Rob Kabacoff, Manning.
4. R For Everyone, Lander, Second Edition, Pearson.
5. The Art Of R Programming, Norman Matloff, No Starch Press.

Web Links:

1. https://onlinecourses.nptel.ac.in/noc19_ma33/preview
2. https://www.coursera.org/projects/getting-started-with-r
3. https://www.geeksforgeeks.org/r-programming-language-introduction/
4. https://www.udacity.com/course/data-analysis-with-r--ud651

APPLICATIONS OF PYTHON-PANDAS

(Skill Oriented Course- II)

(Common to CSE & IT)

		L	T	P	C
IV Semester					
Course Code:		**0**	**0**	**4**	**2**

Course Objectives:

COB 1: To understand the fundamentals of the Pandas library in Python
COB 2: To develop skills of handling data with pandas library

COB 3: To develop basic skills in data analysis and visualization

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Use Pandas to create and manipulate data structures like Series and DataFrames.
CO 2: Experiment with arrays, queries, and dataframes.
CO 3: Apply dataframe structures for cleaning and processing and manipulating files
CO 4: Understand best practices for creating basic charts

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO1(K3)	3	2	1	1	3	-	-	-	-	-	-	2
CO2(K3)	3	3	3	3	3	-	-	-	-	-	-	2
CO3(K3)	3	2	1	1	3	-	-	-	-	-	-	2
CO4(K3)	3	2	1	1	3	-	-	-	-	-	-	2

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1(K3)	PSO2(K3)	PSO3(K4)
CO1(K3)	3	2	3
CO2(K3)	3	2	3
CO3(K3)	3	2	3
CO4(K3)	3	2	3

Perform the following:

1) Pandas Installation
2) Creating DataFrames

Exercises:

A) Pandas Data Series:

1) Write a Pandas program to create and display a one-dimensional array-like object containing an array of data using Pandas module.
2) Write a Pandas program to convert a Panda module Series to Python list and it's type.

3) Write a Pandas program to add, subtract, multiple and divide two Pandas Series.

4) Write a Pandas program to convert a NumPy array to a Pandas series.
Sample Series: NumPy array:
[10 20 30 40 50]
Converted Pandas series: 0 10
1 20
2 30
3 40
4 50
D type: int64

B) Pandas Data Frames:

Consider Sample Python dictionary data and list labels:
exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

1) Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.
2) Write a Pandas program to change the name 'James' to 'Suresh' in name column of the DataFrame.
3) Write a Pandas program to insert a new column in existing DataFrame.
4) Write a Pandas program to get list from DataFrame column headers.
5) Write a Pandas program to get list from DataFrame column headers.

C) Pandas Index:

1) Write a Pandas program to display the default index and set a column as an Index in a given dataframe.
2) Write a Pandas program to create an index labels by using 64-bit integers, using floating-point numbers in a given dataframe.

D) Pandas Joining and merging DataFrame:

1) Write a Pandas program to join the two given dataframes along rows and assign all data.
2) Write a Pandas program to append a list of dictionaries or series to a existing DataFrame and display the combined data.
3) Write a Pandas program to join the two dataframes with matching records from both sides where available.

E) Pandas Time Series:

1) Write a Pandas program to create
a) Datetime object for Jan 152012.

b) Specific date and time of 9:20 pm.

c) Local date and time.

d) A date without time.

e) Current date.

f) Time from a datetime.

g) Current local time.

2) Write a Pandas program to create a date from a given year, month, day and another date from a given string formats.
3) Write a Pandas program to create a time-series with two index labels and random values. Also print the type of the index.

F) Pandas Grouping Aggregate:
Consider dataset:

	school	Class	name	date_Of_Birth	age	height	weight	Address
S1	s001	V	Alberto Franco	15/05/2002	12	173	35	street1
S2	s002	V	Gino Mcneill	17/05/2002	12	192	32	street2
S3	s003	VI	Ryan Parkes	16/02/1999	13	186	33	street3
S4	s001	VI	Eesha Hinton	25/09/1998	13	167	30	street1
S5	s002	V	Gino Mcneill	11/05/2002	14	151	31	street2
S6	s004	VI	David Parkes	15/09/1997	12	159	32	street4

1) Write a Pandas program to split the following data frame into groups based on school code. Also check the type of Group By object.
2) Write a Pandas program to split the following data frame by school code and get mean, min, and max value of age for each school.

G) Pandas Styling:
1) Create a data frame of ten rows, four columns with random values. Write a Pandas program to highlight the negative numbers red and positive numbers black.
2) Create a data frame of ten rows, four columns with random values. Write a Pandas program to highlight the maximum value in each column.
3) Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight dataframe's specific columns.

H) Excel:
1) Write a Pandas program to import excel data into a Pandas data frame.
2) Write a Pandas program to find the sum, mean, max, min value of a column of file.

I) Plotting:
1) Write a Pandas program to create a horizontal stacked bar plot of opening,

closing stock prices of any stock dataset between two specific dates.

2) Write a Pandas program to create a histograms plot of opening, closing, high, low stock prices of stock dataset between two specific dates.
3) Write a Pandas program to create a stacked histograms plot of opening, closing, high, low stock prices of stock dataset between two specific dates with more bins.

Reference Books:

1. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython by Wes Mc Kinney, O'Reilly Media.
2. Learning the Pandas Library, Matt Harrison.
3. Hands-On Data Analysis with NumPy and Pandas, Curtis Miller
4. Pandas for Everyone: Python Data Analysis Daniel Y. Chen.

Web Links:

1. https://pandas.pydata.org/
2. https://www.w3schools.com/python/pandas/default.asp
3. https://www.tutorialspoint.com/python_pandas/index.htm

WEB APPLICATION DEVELOPMENT USING FULL STACK FRONTEND DEVELOPMENT – MODULE – 2

(Skill Oriented Course- II)
(Common to CSE & IT)

	L	T	P	C
IV Semester	**L**	**T**	**P**	**C**
Course Code:	**0**	**0**	**4**	**2**

Course Objectives:

COB 1: To illustrate the basics of JavaScript and its importance in web pages.
COB 2: To embed JavaScript in a web page.
COB 3: To build dynamic web pages using JavaScript.
COB 4: To facilitate the usage of objects and events in web pages using JavaScript.
COB 5: To experiment with JavaScript for validating the content in HTML form fields.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Build a web page by embedding JavaScript to invoke programming ability.
CO 2: Make use of Pre-defined JavaScript objects properties and methods.
CO 3: Experiment with JavaScript to develop dynamic web pages and validate forms.
CO 4: Choose the appropriate properties and methods to design custom objects.
CO 5: Develop applications using the event handling in JavaScript.

Mapping of course outcomes with program outcomes:

CO/PO	PO 1 (K3)	PO 2 (K4)	PO 3 (K5)	PO 4 (K5)	PO 5 (K3)	PO 6 (K3)	PO 7 (K2)	PO 8 (K3)	PO 9 (K2)	PO 10 (K2)	PO11 (K3)	PO12 (K1)
CO 1(K3)	2	3	3	-	3	-	-	-	3	-	-	-
CO 2(K3)	2	3	3	-	3	-	-	-	3	-	-	-
CO 3(K3)	2	3	3	3	3	-	-	-	3	-	-	-
CO 4(K3)	2	3	3	3	3	-	-	-	3	-	-	3
CO 5(K3)	2	3	3	3	3	-	-	-	3	-	-	3

Mapping of course outcomes with program Specific Outcomes:

CO/PSO	PSO1(K3)	PSO2(K3)	PSO3(K4)
CO 1(K3)	2	-	-
CO 2(K3)	2	-	-
CO 3(K3)	2	-	-
CO 4(K3)	2	3	-
CO 5(K3)	2	3	-

Perform Experiments Related to The Following Concepts:

JavaScript:

Introduction to JavaScript, Applying JavaScript - internal and external, Understanding JS Syntax, Introduction to document and window Object, Variables and Operators, Data Types, Pop-up boxes, Input and Output, Num Type Conversion, Math and String

Manipulation, Arrays, Date and Time, Conditional Statements, Switch Case, Looping in JS, Functions, Objects, Events.

List of Experiments:

Experiment 1: **Introduction to JavaScript, Applying JavaScript - internal and external, Understanding JS Syntax**

f. Explain the importance of JavaScript.
g. What JavaScript can do?
h. Write a program to explain how to use JavaScript in a web page.
i. Write a program to explain how to link an external JavaScript page to a HTML page.

Experiment 2: **Introduction to document object**

d. Explain JavaScript document object with properties and methods.
e. Write a JavaScript program to explain the usage of Document object properties.
i. document ii. length iii. name
iv. parent v. status vi. screenX, screenY
f. Write a JavaScript program to explain the usage of Document object methods.
i. open() ii. close()
iii. write() &writeln() iv. getElementById()
v. getElementByName() vi. GetElementsByTagName()

Experiment 3: **Introduction to window object**

a. Explain JavaScript window object with properties and methods.
b. Write a JavaScript program to explain the usage of window object properties.
i. title ii. url iii. cookie
iv. last modified v. domain vi. readyState
c. Write a JavaScript program to explain the usage of Document object methods.
i. open() ii. close() iii. print()
iv. stop() v. focus() vi. getInterval(), setInterval()

Experiment 4: **Variables and Operators, Data Types and Pop-up boxes**

c. Write a JavaScript program to explain different types of variable. Write the differences between variables created with var, let, const keywords.
d. Write a JavaScript program to explain data types with example program.
e. Write a program to explain the Pop-up boxes in JavaScript. (prompt box, alert box and confirm box).
f. Create a webpage which uses prompt dialogue box to ask a user for their name, age and salary. Display the information they enter on the page formatted as a small table.

Experiment 5: **Input and Output statements, Num Type Conversion**

c. Write a JavaScript program to explain the different ways for displaying output.

d. Write a JavaScript program to explain the different ways for taking input.

e. Write a program that uses JavaScript that adds some numbers together using number conversion, concatenates a couple of strings and then shows the result in an alert dialogue box and on the page. (Hint: Use parseInt() for converting string input to integer to add 2 integers)

Experiment 6: **Math and String Manipulation**

a. Write a JavaScript program to explain the use of a Math object.

i. properties - PI, SQRT2

ii. functions – round(), ceil(), floor(), trunc(), random(), max(), min(), pow(), sqrt(), parseInt(), parseFloat()

b. Write a JavaScript program to explain the use of String object.

i. properties - length

ii. functions – charAt(), concat(), indexOf(), lastIndexOf(), split(), trim(), slice(), substr(), substring(), toLowercase(), toUppercase(), replace().

Experiment 7: **Arrays, Date and Time**

c. Write a program to explain the concept of Arrays. How to create, access an array, adding elements to array, searching element in an array, removing array numbers and sorting the numbers .

d. Write a JavaScript program to explain about Array object.

i. properties - length

e. functions - concat(), join(), pop(), push(), reverse(), shift(), slice(), sort(), splice(), unshift()

f. Write a JavaScript program to explain about Date object.

Experiment 8: **Conditional Statements, switch**

a. Write a program to demonstrate the conditional statements – if, if else, else if ladder.

b. Write a script that reads an integer and determines and displays whether it is an odd or even number.

c. Write a java script which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words “LARGER NUMBER” in an information message dialog. If the numbers are equal, output HTML text as “EQUAL NUMBERS”.

d. Write a JavaScript program to display week days using switch case.

Experiment 9: **Loops**

a. Write a JavaScript program to print 1 to 10 numbers using for, while and do-while loops.

b. Write a JavaScript program to print data in object using for-in, for-each and for-of loops

c. Develop a java script to determine whether a given number is an ‘ARMSTRONG NUMBER’ or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e.,13 + 53+ 33 = 153]

Experiment 10: **Functions and Objects**

a. Write a program to explain the concept of functions. Define a function, pass parameters, return values, local and global scope.
b. Design a appropriate JavaScript function should be called to display
    i. Factorial of that number
    ii. Fibonacci series up to that number
    iii. Prime numbers up to that number
    iv. Is it palindrome or not

Experiment 11: **Objects**

Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

Experiment 12: **Events**

a. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate java script function should be called to display
    i. Factorial of that number
    ii. Fibonacci series up to that number
    iii. Prime numbers up to that number
    iv. Is it palindrome or not
b. Write a JavaScript to validate the following fields in a registration page created in Experiment 2
    i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
    ii. Mobile (only numbers and length 10 digits)
    iii. E-mail (should not contain invalid email addresses)

List of Augmented Experiments: (Weeks 13 – Week 16)

(Any two of the following experiments can be performed)

13. Write a JavaScript that takes a number from one text field in the range of 0-999 and display it in other text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.
14. Write a Java Script to display the denomination of the amount deposited in the bank in terms of 100’s, 50’s, 20’s, 10’s, 5’s, 2’s & 1’s. (Eg: If deposited amount is Rs.163, the output should be 1-100’s, 1-50’s, 1- 10’s, 1-2’s & 1-1’s)
15. Write a Java Script to validate all the fields in a registration form.
16. Write JavaScript programs on Event Handling
    i. Open a Window from the current window.
    ii. Change color of background at each click of button or refresh of a page.
    iii. On Mouse over event.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson.
2. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, Dream Tech.
3. An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sand a S Katila, Cengage Learning.

Web Links

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css

***