RAMAIAH
Institute of Technology

CURRICULUM

Academic Year 2026– 2027

Batch - 2025

III & IV Semester B. E.

INFORMATION SCIENCE AND ENGINEERING

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute, Affiliated to VTU)
(Approved by AICTE, New Delhi & Govt. of Karnataka)
Accredited by NBA & NAAC with A^+ Grade

About the Institute:

Dr. M. S. Ramaiah a philanthropist, founded 'Gokula Education Foundation' in 1962 with an objective of serving the society. M S Ramaiah Institute of Technology (MSRIT) was established under the aegis of this foundation in the same year, creating a landmark in technical education in India. MSRIT offers 18 UG programs and 15 PG programs. All these programs are approved by AICTE. All eligible UG and PG programs are accredited by National Board of Accreditation (NBA). The institute is accredited with **'A^{+}' grade by NAAC in March 2021** for 5 years. University Grants Commission (UGC) & Visvesvaraya Technological University (VTU) have conferred Autonomous Status to MSRIT for both UG and PG Programs since 2007. The institute has also been conferred autonomous status for Ph.D. program since 2021. The institute is a participant to the Technical Education Quality Improvement Program (TEQIP), an initiative of the Government of India. The institute has 380 competent faculty out of which 70% are doctorates. Some of the distinguished features of MSRIT are: State of the art laboratories, individual computing facility for all faculty members, all research departments active with sponsored funded projects and more than 300 scholars pursuing Ph.D. To promote research culture, the institute has established Centre of Excellence for Imaging Technologies, Centre for Advanced Materials Technology, Centre for Antennas and Radio Frequency systems (CARFS), Center for Cyber Physical Systems, Schneider Centre of Excellence & Centre for Bio and Energy Materials Innovation.

The Entrepreneurship Development Cell (EDC) and "Ramaiah Evolute" have been set up on campus to incubate startups. MSRIT has a strong Placement and Training department with a committed team, a good Mentoring/Proctorial system, a fully equipped Sports department, large air-conditioned library with good collection of book volumes and subscription to International and National Journals. The Digital Library subscribes to online e-journals from Elsevier Science Direct, IEEE, Taylor & Francis, Springer Link, etc. The Institute is a member of DELNET, CMTI and VTU E-Library Consortium. The Institute has a modern auditorium, recording studio, and several hi-tech conference halls with video conferencing facilities. The institute has excellent hostel facilities for boys and girls. MSRIT Alumni have distinguished themselves by occupying high positions in India and abroad and are in touch with the institute through an active Alumni Association.

As per the National Institutional Ranking Framework (NIRF), MoE, Government of India, M S Ramaiah Institute of Technology has achieved 75th rank among 1584 top Engineering Institutions & 31st Rank among 131 Schools of Architecture in India for the year 2025.

About the Department:

The Department of Information Science and Engineering (ISE) was established in the year 1992 with an objective of producing high quality professionals to meet the demands of the emerging field of Information Technology. Department offers Bachelor's program in Information Science and Engineering (B. E), Master's program in Data Science (MTech) and Doctoral program (Ph.D.). The Department of Information Science and Engineering, is a progressive department that has made significant contributions to Academics, Research and Innovation. Under Graduate (UG) is accredited by the National Board of Accreditation in 2001, 2004, 2010, 2015, 2018 and reaccredited in 2022 under Tier-1 till 2028. The department has highly qualified and competent faculty members committed to innovative teaching learning and quality research. Department has 8 well-equipped state of the art laboratories which meets the requirement of curriculum, innovation and research. Collaboration with industries such as Apple, Unisys, Mindtree, Intel, Google, SECO, IBM, NVIDIA etc, has a significant impact on the curriculum, computing infrastructure, teaching & learning and research. The curriculum is centered around Data Science, Artificial Intelligence, IOT, Cloud & Distributed Computing, System Programming, Computer Security and Software development. Curriculum and the teaching learning process ensure that the students demonstrate technical competence, ethical reasoning, creativity in identification & formulation of the problems and develop solutions by using appropriate tools & techniques. Department has established technical clubs/ professional student chapters to provide collaborative learning platform for the students. Echo system has been built to initiate start-ups/Innovation at the department level along with the mentorship program. The activities of the Department led to high profile placements, motivation to become an entrepreneur, and encouragement for higher learning.

VISION OF THE INSTITUTE

To be an Institution of International Eminence, renowned for imparting quality technical education, cutting edge research and innovation to meet global socio-economic needs

MISSION OF THE INSTITUTE

MSRIT shall meet the global socio-economic needs through

1. Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization
2. Establishing research clusters in emerging areas in collaboration with globally reputed organizations
3. Establishing innovative skills development, techno-entrepreneurial activities and consultancy for socio-economic needs

QUALITY POLICY

We at MS Ramaiah Institute of Technology strive to deliver comprehensive, continually enhanced, global quality technical and management education through an established Quality Management System complemented by the synergistic interaction of the stake holders concerned

VISION OF THE DEPARTMENT

To evolve as an outstanding education and research center of Information Technology to create high quality Engineering Professionals for the betterment of Society

MISSION OF THE DEPARTMENT

Department of Information Science and Engineering shall create high quality IT Engineering Professionals for the betterment of society by:

1. Providing education through an ever-improving curriculum and effective pedagogy techniques.
2. Encouraging extra and co-curricular activities to develop their overall personality along with technical skills.
3. Collaborating with industry and academia for strengthening research, innovation and entrepreneurship ecosystem.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs):

PEO1: Become competent Information Technology professionals with continuous progress in career or learning.

PEO2: Productively engage with society by practicing research or entrepreneurship.

PEO3: Function effectively as professionals in a team environment or individually.

KNOWLEDGE AND ATTITUDE PROFILE (WK)

WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

WK2: Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.

WK3: A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.

WK4: Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.

WK5: Knowledge, including efficient resource use, environmental impacts, whole-life cost, reuse of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.

WK6: Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.

WK7: Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.

WK8: Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.

WK9: Ethics, inclusive behaviour and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

PROGRAM OUTCOMES (POs):

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WKS)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES (PSOs):

PSO1: Apply Mathematical models, programming paradigms and software development practices to solve real world problems

PSO2: Adopt computing and communication models for developing IT solutions.

PSO3: Acquire data engineering skills to develop intelligent systems in a multidisciplinary environment.

Semester wise Credit Breakdown for B.E Degree Curriculum
Batch 2025-29

Semester / Course Category	First	Second	Third	Fourth	Fifth	Sixth	Seventh	Eighth	Total Credits
Applied Science Course (ASC)	08	08	03	03	--	--	--	--	**22**
Engineering Science Course (ESC)	09	11	--	--	--	--	--	--	**20**
Humanities, Social Sciences and Management (HSMC)	02	-	--	--	03	--	03	--	**08**
Ability Enhancement Course (AEC)	01	01	01	01	01	01	--	--	**06**
Universal Human Values (UHV)	--	--	02	--	--	--	--	--	**02**
Professional Core Courses (PCC)	--	--	11	09	11	11	04	--	**46**
Professional Core Course (IC)	--	--	04	04	04	--	04	--	**16**
Professional Elective Course (PEC)	--	--	--	03	03	03	03	--	**12**
Open Elective Course (OEC)	--	--	--	--	--	03	03	--	**06**
Internship (INT)	--	--	--	00	--	--	--	04	**04**
Mini Project / Project Work (PW)	--	--	--	--	--	04	02	12	**18**
Non Credit Mandatory Courses (NCMC)	--	--	Yes	Yes	Yes	--	--	--	**--**
Total Credits	**20**	**20**	**21**	**20**	**22**	**22**	**19**	**16**	**160**

SCHEME OF TEACHING III SEMESTER (2025 Batch)

B.E. in Information Science and Engineering
Scheme of Teaching – Batch : 2025 – 2029

III SEMESTER

Sl. No.	Subject Code	Subject	Teaching Department	Category	Credits				Total contact hours /week			
					L	T	P	Total	L	T	P	S
1	IS31	Transform Techniques and Linear Algebra	Mathematics	ASC	2	1	0	3	2	2	0	2
2	IS32	Digital Design and Computer Organization	ISE	PCC(IC)	3	0	1	4	3	0	2	3
3	IS33	Data Structures	ISE	PCC	3	0	0	3	3	0	0	3
4	IS34	Object Oriented Programming with Java	ISE	PCC	3	0	0	3	3	0	0	3
5	IS35	Discrete Mathematical Structures	ISE	PCC	3	0	0	3	3	0	0	3
6	UHV36	Universal Human Value Course	ISE	UHV	2	0	0	2	2	0	0	2
7	ISAEC37	Ability Enhancement Course- III	ISE	AEC	1	0	0	1	1	0	0	1
8	ISL38	Data Structures Lab	ISE	PCC	0	0	1	1	0	0	2	0
9	ISL39	Object-Oriented Programming with Java Lab	ISE	PCC	0	0	1	1	0	0	2	0
				Total	**17**	**1**	**3**	**21**	**17**	**2**	**6**	**17**
10	PE69	Physical Education		NCMC	All students have to register compulsorily for any one of the courses with the concerned coordinator (Yoga Teacher/ Physical Education Director/ NSS Coordinator) in the beginning of the III semester. Attending the registered course from III to VI semesters. Successful completion of the registered course and requisite CIE score is mandatory for the award of the degree.							
	YO69	Yoga										
	NS69	NSS										
11	AM31	Additional Mathematics - I *	Mathematics	NCMC	0	0	0	0	-			

Ability Enhancement Course – V

Sl.No	Subject Code	Subject Name
1	ISAEC371	Web Programming
2	ISAEC372	Green IT & Sustainability
3	ISAEC373	Engineering Economics

*** Environmental Studies is under the category of NCMC, 1-hour teaching per week has to be allocated in the time table.**
NOTE: Minimum of 1 subject should have a Tutorial component of 1 Credit.

Nomenclature: IC: Integrated Course (Practical Integrated with Theory Course), **PCC:** Professional Core Course, **PEC:** Professional Elective Course, **AEC:** Ability Enhancement Course**, HSMC:** Humanity, Social Science & Management Course, **NCMC:** Non-credit Mandatory Course

Credit Definition:

- **1 credit = 1 hour/week of lecture (L)**
- **1 credit = 2 hours/week of tutorial (T)**
- **1 credit = 2 hours/week of practical(P)**
- **Self-Learning Hours (S)**

PCC(IC) is an Integrated course, combining theory with practical components.

- Theory sessions shall be conducted for 3 hours per week, while the practical sessions shall be conducted for 2 hours per week.
- Theory components shall be evaluated through both Continuous Internal Evaluation (CIE) and Semester End Examination (SEE).
- The practical component shall be assessed only through CIE (No SEE).
- The CIE shall consist of two internal test of 30 marks each. Each test will have three questions and student shall answer any two full questions. Each question shall have all CO's addressed for that test. An average of two test shall be taken to award 30 marks.
- The CIE for the practical component shall be evaluated for 20 marks. Each laboratory experiment shall be evaluated for 10 marks based on the experiment record, conduction of the experiment, and observations, and the cumulative marks shall be scaled down to 15 marks. Another 05 marks from a lab test and viva.
- The SEE shall be evaluated for 100 marks, which shall be scaled down to 50 marks.
- The SEE examination shall consist of 10 questions with internal choice and student shall answer any five full questions.
- However, questions from the practical component of the Integrated course (IC) can be included in the SEE question paper.

PCC & PEC & HSMC courses are theory courses with or without tutorial sessions.

- Each theory credit corresponds to one contact hour per week, while each tutorial credit corresponds to two contact hours per week.
- The course shall be evaluated through Continuous Internal Evaluation (CIE) and Semester End Examination (SEE), each having a weightage of 50%.
- The CIE is evaluated for 50 marks through internal tests (30 marks) and other components (20 marks). Each internal test will have three questions and students shall answer any two full questions. Each question shall address all the Course Outcomes (COs) prescribed for that test. The average of the two tests shall be considered for awarding 30 marks. The remaining 20 marks shall be assessed through assignments, tutorial tests, quizzes, surprise tests, mini-projects, or other components as decided by the course instructor.

- The SEE shall be evaluated for 100 marks, which shall be scaled down to 50 marks.
- The SEE examination shall consist of ten questions with internal choice (two questions per unit). Students shall answer any five full questions, selecting at least one question from each unit.

AEC is a 1-credit course involving 1 contact hour per week and NCMC is a 0-credit course involving 1 contact hour per week. The courses shall be evaluated through Continuous Internal Evaluation (CIE) and Semester End Examination (SEE), each having a weightage of 50%.

- The CIE will comprise of two tests of 30 marks each having Part-A with MCQs for 6 marks and Part-B with Descriptive questions. The remaining 20 marks shall be assessed through assignments, surprise tests, quizzes, or other components as decided by the course instructor.
- The SEE will comprise of Part-A with MCQs (10 marks) and Part-B (5x8=40 marks) with descriptive questions.

PCC Laboratory is a 1-credit lab course involving 2 hours of lab sessions per week and uses both Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) for assessment.

- CIE totals 50 marks, with 30 marks for lab sessions (Each laboratory experiment shall be evaluated for 10 marks based on the experiment record, conduction of the experiment, and observations, and the cumulative marks shall be scaled down to 30 marks.). Another 15 marks from a lab test and 05 marks for viva.
- SEE totals 50 marks, comprising 40 marks for procedure write up, experiment conduction & results and 10 marks for viva voce.

NCMC is a 0-credit course involving 1 contact hour per week: The students shall attend classes for the course during the semester and complete all formalities of attendance and CIE. The CIE shall be evaluated for 100 marks. In case, any student fails to secure the minimum 40% of the prescribed CIE marks, he/she shall be deemed to have secured 'F' grade. In such a case, the student has to fulfill the requirements during subsequent semester/s to appear for CIE. Incase student fails to register for the said course/ falls short of attendance, he/she will repeat the course whenever it is offered next. This course shall have CIE component only and **no SEE component**. The completion of the course shall be mandatory for the award of the degree.

AICTE Activity Points Requirement for BE Programs: As per AICTE guidelines (refer Chapter 6 – *AICTE Activity Point Program, Model Internship Guidelines*), in addition to academic requirements, students must earn a specified number of **Activity Points** to be eligible for the award of the degree. The points to be earned is:

1. **Regular students** admitted to a 4-year degree program must earn **100 Activity Points**.
2. **Lateral entry students** (joining from the second year) must earn **75 Activity Points**.
3. **Students transferred** from other universities directly into the V semester must earn **50 Activity Points** from the date of entry into VTU.

These Activity Points are non-credit and will not be considered for the SGPA/CGPA or be used for vertical progression. However, earning Activity Points is mandatory for the award of the degree, and the points earned will be reflected on the eighth semester Grade Card. If a student completes all the semesters (eight or six) at the end of the program but fails to earn the required Activity Points, the VIII semester Grade Card will be withheld until the requirement is fulfilled. Also, the degree will be awarded only after the Grade Card has been released. The hours spent earning the activity points will not be counted for regular attendance requirements. Students can accumulate these points at any time during their program period, including weekends, holidays, and vacations, starting from the year of admission, provided they meet the minimum hours of engagement prescribed for each activity by AICTE.

SCHEME OF TEACHING IV SEMESTER

B.E. in Information Science and Engineering - Scheme of Teaching – Batch: 2025												
IV SEMESTER												
Sl. No.	Subject Code	Subject	Teaching Department	Category	Credits				Total contact hours /week			
					L	T	P	Total	L	T	P	S
1	IS41	Microcontrollers	ISE	PCC(IC)	3	0	1	4	3	0	2	3
2	IS42	Probability, Statistics and Queuing Theory	Maths	PCC	2	1	0	3	2	2	0	2
3	IS43	Design and Analysis of Algorithms	ISE	PCC	3	0	0	3	3	0	0	3
4	IS44	Operating Systems	ISE	PCC	3	0	0	3	3	0	0	3
5	ISE45x	Program Elective Course – 1	ISE	PEC	3	0	0	3	3	0	0	3
6	ISAEC46	Ability Enhancement Course – IV	ISE	AEC	1	0	0	1	1	0	0	1
7	ISL47	Data Analytics Lab	ISE	PCC	0	0	1	1	0	0	2	0
8	ISL48	Design and Analysis of Algorithms Lab	ISE	PCC	0	0	1	1	0	0	2	0
9	ISL49	Unix Programming Lab	ISE	PCC	0	0	1	1	0	0	2	0
	Total				**15**	**1**	**4**	**20**	**15**	**2**	**8**	**15**
10	INT410	Inter/ Intra Institutional Internship	ISE	NCMC	0	0	0	0	-			
11	AM41	Additional Mathematics II *	Mathematics	NCMC	0	0	0	0	-			

Programme Electives Course – 1

Sl.No	Subject Code	Subject Name
1	ISE451	Artificial Intelligence
2	ISE452	Operations Research
3	ISE453	Cyber Security and Ethical Hacking

Ability enhancement Course – IV

Sl.No	Subject Code	Subject Name
1	ISAEC461	UI / UX
2	ISAEC462	GNU Linux System Administration[
3	ISAEC463	No SQL Database

NOTE: Minimum of 1 subject should have a Tutorial component of 1 Credit.

Nomenclature: PCC: Professional Core Course, **PEC:** Professional Elective Course, **SDC:** Skill Development Course**, OEC:** Open Elective Course**, PW:** Project Work, **NCMC:** Non-credit Mandatory Course.

Credit Definition:

- **1 credit = 1 hour/week of lecture (L)**
- **1 credit = 2 hours/week of tutorial (T)**
- **1 credit = 2 hours/week of practical(P)**
- **Self-Learning Hours (S)**

PCC & PEC & OEC courses are theory courses with or without tutorial sessions.

- Each theory credit corresponds to one contact hour per week, while each tutorial credit corresponds to two contact hours per week.
- The course shall be evaluated through Continuous Internal Evaluation (CIE) and Semester End Examination (SEE), each having a weightage of 50%.
- The CIE is evaluated for 50 marks through internal tests (30 marks) and other components (20 marks). Each internal test will have three questions and students shall answer any two full questions. Each question shall address all the Course Outcomes (COs) prescribed for that test. The average of the two tests shall be considered for awarding 30 marks. The remaining 20 marks shall be assessed through assignments, tutorial tests, quizzes, surprise tests, mini-projects, or other components as decided by the course instructor.
- The SEE shall be evaluated for 100 marks, which shall be scaled down to 50 marks.
- The SEE examination shall consist of ten questions with internal choice (two questions per unit). Students shall answer any five full questions, selecting at least one question from each unit.

SDC Courses:

- The Technical Skill Enhancement course is designed on the Finishing School model, providing students with an industry-oriented overview and integration of the technical concepts studied from the I to VI semesters.

- Industry experts are invited to deliver the course during the semester, with **two-hour sessions held every week**, focusing on the application-oriented coverage of core technical concepts from the I to VI semesters.
- Assessment shall be based on CIE for 100 marks. Marks shall be awarded for each session by the **internal faculty coordinator** for attendance and performance in quiz/assignment/ capstone projects/session-specific activity designed. Ten session specific activities to be designed for 10 marks each.
- The Technical Skill Enhancement course shall have CIE component only and **no SEE component.**

Mini-project work Assessment (PW):

- Mini Project is a laboratory-oriented course which will provide a platform to students to enhance their practical knowledge and skills by the development of small systems/applications.
- Based on the ability/abilities of the student/s and recommendations of the mentor, a single discipline or a multidisciplinary Mini- project can be assigned to a group having not more than 4 students.
- A guide/s shall be allocated to the group in the beginning of the semester.
- The CIE totals 50 marks and shall be awarded by a committee constituted by of the Head of the concerned department. The CIE marks shall be awarded with 20 marks for the first presentation and 30 marks for the second presentation as per the rubrics defined by the department.
- SEE shall be conducted for 50 marks by two examiners appointed from within the Institute/ Department. SEE marks awarded for the mini project shall be based on the evaluation of brief outline of the work, presentation & report for 40 marks and 10 marks question & answer session.

PCC Laboratory is a 1-credit lab course involving 2 hours of lab sessions per week and uses both Continuous Internal Evaluation (CIE) and Semester End Examination (SEE) for assessment.

- CIE totals 50 marks, with 30 marks for lab sessions (Each laboratory experiment shall be evaluated for 10 marks based on the experiment record, conduction of the experiment, and observations, and the cumulative marks shall be scaled down to 30 marks.). Another 15 marks from a lab test and 05 marks for viva.
- SEE totals 50 marks, comprising 40 marks for procedure write up, experiment conduction & results and 10 marks for viva voce.

AICTE Activity Points Requirement for BE Programs: As per AICTE guidelines (refer Chapter 6 – *AICTE Activity Point Program, Model Internship Guidelines*), in addition to academic requirements, students must earn a specified number of **Activity Points** to be eligible for the award of the degree. The points to be earned is:

1. **Regular students** admitted to a 4-year degree program must earn **100 Activity Points**.
2. **Lateral entry students** (joining from the second year) must earn **75 Activity Points**.
3. **Students transferred** from other universities directly into the V semester must earn **50 Activity Points** from the date of entry into VTU.

These Activity Points are non-credit and will not be considered for the SGPA/CGPA or be used for vertical progression. However, earning Activity Points is mandatory for the award of the degree, and the points earned will be reflected on the eighth semester Grade Card. If a student completes all the semesters (eight or six) at the end of the program but fails to earn the required Activity Points, the VIII semester Grade Card will be withheld until the requirement is fulfilled. Also, the degree will be awarded only after the Grade Card has been released. The hours spent earning the activity points will not be counted for regular attendance requirements. Students can accumulate these points at any time during their program period, including weekends, holidays, and vacations, starting from the year of admission, provided they meet the minimum hours of engagement prescribed for each activity by AICTE.

The Non Credit Mandatory Course, Physical Education (Sport and Athletics)/Yoga/National Service Scheme (NSS):

1. Student shall select any one of the NCMC's namely, Physical Education (Sport and Athletics)/Yoga/ NSS prescribed for four semesters and shall attend the course from the III semester to VI semester to complete all the formalities of the course and earn requisite CIE score. The Pass/Fail shall be reflected in the VI semester grade card.
2. The above mentioned NCMC's shall not be considered for vertical progression as well as for the calculation of SGPA/CGPA but completion of the courses shall be mandatory for the award of degree.
3. CIE marks will be allotted by the concerned course teacher based on attendance and performance in the practice sessions/field work every semester (III to VI) for 100 marks and the total 400 marks scored by the student are scaled down to 100 in the VI semester.
4. In case, any student fails to secure the minimum 40% of the prescribed marks, he/she shall be deemed to have secured 'F' grade.

Transform Techniques and Linear Algebra	
Course Code: IS31	**Credits: 2:1:0:2**
Prerequisites: Calculus and Basics of Linear Algebra	**Contact Hours: 30L + 15T + 30S**
Course Coordinator: Dr. Govindaraju M V	

Course Content

Unit I

Laplace Transforms: Definition, transforms of elementary functions, properties of Laplace transforms, existence conditions, transform of derivatives, integrals, multiplication by t^n, division by t, evaluation of integrals by Laplace transforms.

Pedagogy / Course delivery tools: Chalk and talk.

Links: https://nptel.ac.in/courses/111/105/111105134/

Unit II

Application of Laplace Transforms: Laplace transform of periodic function. Unit–step function, unit–impulse function. inverse transforms, convolution theorem. Solution of linear differential equations and simultaneous linear differential equations using Laplace transforms.

Pedagogy / Course delivery tools: Chalk and talk.

Links: https://nptel.ac.in/courses/111/105/111105134/

Unit III

Z transforms: Z-transform of Standard functions, Properties of Z-transforms, Convergence of Z-transforms, Inverse Z-transform, Application of Z-transforms to solve difference equations, linear recurrence relations.

Pedagogy / Course delivery tools: Chalk and talk.

Links: https://nptel.ac.in/courses/111106111
https://a.impartus.com/ilc/#/course/2975585/1331

Unit IV

Orthogonal Projections: Review of vector space, complement subspaces, orthogonality of four fundamental subspaces and its dimensions, QR-factorization, the complete solution to Ax = b, least-squares approximations, constraint optimization. Quadratic forms, positive definite matrices.

Pedagogy / Course delivery tools: Chalk and talk.

Links: https://nptel.ac.in/courses/106108482
https://web.mit.edu/18.06/www/

Unit V

Applications of Eigenvalue Decomposition: Review of eigenvalues and eigenvectors. Symmetric matrices, complex matrices, Hermitian and unitary matrices, similarity and orthogonal diagonalization, the singular value decomposition (SVD), principal component analysis (PCA), applications to linear recurrence relations, Fibonacci sequence, stability of the differential equations.

Pedagogy / Course delivery tools: Chalk and talk.

Links: https://nptel.ac.in/courses/106108482
https://web.mit.edu/18.06/www/

Textbooks

1. Kreyszig, E. (2025). Advanced Engineering Mathematics (11th ed.). Wiley-India Publishers.
2. Lay, D. C., Lay, S. R., & McDonald, J. J. (2021). Linear Algebra and its Applications (6th ed.). Pearson.
3. Strang, G. (2016). Linear Algebra and its Applications (5th ed.).

Reference Books:

1. O'Neil, P. V. (2023). Advanced Engineering Mathematics (8th ed.). Cengage Learning.
2. Williams, G. (2019). Linear Algebra with Applications (9th ed.). Jones and Bartlett Press.
3. Grewal, B. S. (2024). Higher Engineering Mathematics (45th ed.). Khanna Publishers.

Self- Learning:

Understand the real world applications of Laplace transforms in electrical circuits, Z transforms in solving difference equations / recurrence relations and Eigen value decomposition to principal component analysis

Course Outcomes

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

1. Apply the properties of Laplace transforms to evaluate integrals and transform elementary functions. (PO: 1, 2 ; PSO:1, 3)
2. Solve linear and simultaneous differential equations by employing inverse Laplace transforms, convolution. **(**PO: 1, 2 ; PSO: 1, 3)
3. Utilize Z-transforms and their properties to analyze convergence and solve linear recurrence relations and difference equations. **(**PO: 1, 2 ; PSO: 1, 3)
4. Analyze vector spaces through orthogonal projections, QR-factorization, least-squares approximations, and the optimization of quadratic forms. (PO: 1, 2 ; PSO: 1, 3)
5. Construct solutions for real-world systems using eigenvalue decomposition, Singular Value Decomposition (SVD), and Principal Component Analysis (PCA). (PO:1, 2; PSO:1, 3)

Tutorial Classes

Session No.	Topic	No. of hours
1	Problems on finding Laplace transforms of elementary functions (e^{at}, $\sin at$, t^n, etc.) using definitions, linearity, and first shifting properties. Evaluating the existence conditions of the transform	2 hours
2	Solving problems involving the Laplace transform of derivatives and integrals, multiplication by t^n, and division by t	2 hours
3	Utilizing Laplace transform properties to compute improper definite integrals of the form $\int_0^\infty f(t)\,dt$ without finding the antiderivative directly.	2 hours

4	Finding the Laplace transform of periodic functions, the unit-step (Heaviside) function, and the unit-impulse (Dirac delta) function. Applying the Convolution Theorem to evaluate inverse transforms.	2 hours
5	Solving first and second-order ordinary linear differential equations with constant coefficients and specified initial conditions using Laplace transforms.	2 hours
6	Solving systems of simultaneous linear differential equations encountered in interconnected physical or electrical engineering systems.	2 hours
7	Computing the Z-transform of standard discrete sequences ($a^n, u[n]$, etc.) and analyzing their properties, with a specific focus on mapping the Region of Convergence.	2 hours
8	Finding the inverse Z-transform using power series expansion, partial fraction expansion, and the residue/convolution methods	2 hours
9	Applying Z-transforms to solve linear difference equations and linear recurrence relations with given initial boundary states.	2 hours
10	Determining the bases, dimensions, and orthogonality of the four fundamental subspaces of a matrix A	2 hours
11	Performing Gram-Schmidt orthogonalization to compute the QR-factorization of a matrix. Finding the complete solution to linear systems.	2 hours
12	Solving overdetermined systems using least-squares approximations, solving constrained optimization problems, and testing matrices for positive definiteness via quadratic forms.	2 hours
13	Review of eigenvalues/eigenvectors. Diagonalizing symmetric and complex matrices (Hermitian, Unitary, and Similarity transformations).	2 hours
14	Step-by-step computation of the Singular Value Decomposition ($A = U\Sigma V^T$) of rectangular matrices and applying it conceptually to Principal Component Analysis (PCA)	2 hours
15	Applying eigenvalue techniques to solve linear recurrence equations and analyzing the stability of systems of differential equations	2 hours

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes addressed**
Internal test – I	30	CO1,CO2,CO3
Internal test – II	30	CO3,CO4,CO5
Average of the two internal tests shall be taken for 30 marks.		
Other components	**Marks**	**Course outcomes addressed**
Quiz	10	CO1,CO2,CO3
Assignment	10	CO3,CO4,CO5
Semester End Examination (SEE)		
Course end examination(Answer any one question from each unit – Internal choice)	100	CO1,CO2,CO3,CO4,CO5

Digital Design and Computer Organization	
Course Code: IS32	**Credits: 3:0:1:3**
Pre – requisites: Basic Electronics	**Contact Hours: 45L+30P+45S**
Course Coordinator: Dr. Anitha P	

Course Content

Unit I

Introduction to Digital System: Introduction, The Map Method for simplifying Boolean expressions, Two, Three and Four-Variable Map, Don't-Care Conditions, Quine-McCluskey Tabular Method, NAND and NOR Implementation, Exclusive-OR Function, Hardware Description Language – Verilog Model of a simple circuit.

- **Pedagogy/Course Delivery tools:** Chalk and talk, Problem solving, use of simulators.
- **Link**: https://onlinecourses.nptel.ac.in/noc24_cs61
- **Impartus recording**: https://a.impartus.com/login/#/

Unit II

Combinational Logic: Introduction, Design Procedure, Binary Adders, BCD Adder, Decoders, Encoders, Multiplexers and Demultiplexers. HDL Models of Combinational Circuits. **Sequential Logic**: Introduction, Sequential Circuits, Storage Elements: Latches, Flip-Flops. Synchronous and Asynchronous counters. HDL codes for Sequential logic Circuits.

- **Pedagogy/Course Delivery tools:** Chalk and talk, Problem solving, use of simulators.
- **Link**: https://edurev.in/p/155137/Combinational-Circuits
- **Link**: https://dec-iitkgp.vlabs.ac.in/exp/sequential-circuits/theory.html

Unit III

Basic Structure of Computers: Functional Units, Basic Operational Concepts, Bus structure, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction sequencing, Addressing Modes.

- **Pedagogy/Course Delivery tools**: Chalk and talk, Power Point Presentation
- **Link**: https://onlinecourses.nptel.ac.in/noc20_cs64

Unit IV

Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access: Bus Arbitration, The memory System: Basic Concepts, Speed, size and Cost of memory systems. Cache Memories – Mapping Functions.

- **Pedagogy/Course Delivery tools**: Chalk and talk, Power Point Presentation
- **Link**: https://onlinecourses.nptel.ac.in/noc20_cs64

Unit V

Basic Processing Unit: Some Fundamental Concepts: Register Transfers, Performing ALU operations, fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction. Pipelining: Basic concepts, Role of Cache memory, Pipeline Performance.

- **Pedagogy/Course Delivery tools**: Chalk and talk, Power Point Presentation

- **Link**: https://onlinecourses.nptel.ac.in/noc20_cs64

List of Simulation Programs:

1. Verilog codes for combinational logic circuits.
2. Verilog program for Adder and Subtractor.
3. Verilog program for Decoders and Encoders.
4. Verilog program for Multiplexer and Demultiplexer.
5. Verilog program for flip flops.
6. Verilog program for asynchronous counter.
7. Instruction Execution using Marie Simulator.
8. ALU design using Logisim simulator.
9. Memory operation using Logisim simulator.
10. Pipelining demonstration using CPU-OS simulator.

Text Book:

1. M. Morris Mano & Michael D. Ciletti, Digital Design with an Introduction to Verilog Design, 6e, Pearson Education,2018.
2. Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5th Edition, Tata McGraw Hill. 2002

Reference Books:

1. Donald P Leach, Albert paulMalvino and Goutam Saha, Digital Principles and Applications, 8th Edition, 2024, Tata McGraw Hill.
2. Stephen Brown,ZvonkoVranesic, Digital Logic Design with VHDL, Third Edition, 2008, Tata McGraw Hill.

Self-Learning: Programming Assignment

Unit No.	Assignment Problems
I	**For the following scenarios build the logic circuits (Separately)** 1. An AI chatbot adjusts its response tone based on user mood and time of day. 2. An AI sensor system monitors four safe conditions 3. Build an alert system using NOR gates 4. An AI compares two 4-bit patterns for image recognition
II	**For the following scenarios build the logic circuits (Separately)** 1. An AI selects one of the AI decision models based on weather conditions 2. An AI robot receives 3-bit code input to select any one of its 6 actions 3. An AI system generates compact binary code for multiple signal inputs 4. Build a digital system that samples a data every 8 seconds AI scheduler rotates between its 5 tasks in cycle
III	**Calculate performance** 1. An edge device runs a ML model for real-time object detection 2. Two processors run the same ML model but have different clock rates
IV	**Design a block diagram** 1. An embedded AI device receives inputs from camera and microphone at the same time An LLM deployed on a multi-level memory accesses matrix data and user input data frequently from the memory (Hint use Cache memory)
V	**Write the register transfer sequence when:**

	A CPU executes a matrix multiplication for a single layer of a neural network.

Course Outcomes(COs):

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

1. Apply various simplification techniques to realize digital circuits using basic and universal gates. (PO1,2,3,5,9,10 & PSO2)
2. Design different types of combinational and sequential circuits to develop digital system. (PO1,2,3,4,5,9,10, PSO2)
3. Analyze machine instructions and addressing modes for evaluating processor performance. (PO1, PO5 & PSO2)
4. Understand the various methods used for communication between processor and I/O devices for data and control flow. (PO1, PO5 & PSO2)
5. Analyse internal Organization of Main Memory/Cache to evaluate Processor Performance. (PO1,2,5 & PSO2)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab programs, Weekly Evaluation and Record	10	CO1, CO2, CO3, CO4,CO5
Lab test	10	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

DATA STRUCTURES	
Course Code: IS33	**Credits: 3:0:0:3**
Pre – requisites: Fundamentals of Computing	**Contact Hours: 45L + 45S**
Course Coordinator: Dr. Kusuma S	

Course Content

Unit I

Introduction to Data Structures: Definition, Classification, Pointers and Dynamic Memory Allocation, Review of Arrays, Representation of Linear Arrays in Memory, Dynamic Allocated Arrays, Structures, Self-Referential Structures, Polynomial Representation Using Arrays.

The Stack: Definition, Representation, ADT, Basic Operations of Stack and Its implementation, Stacks using Dynamic Arrays.

Applications of Stack: Conversion from Infix to Postfix, Evaluation of Postfix Expression.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https://nptel.ac.in/courses/106106133
  https://nptel.ac.in/courses/106102064
  https:// nptel.ac.in/courses/106106127

Unit II

Recursion: Definition, Process and Programming Examples (Tower of Hanoi, Factorial, Binary Search), Role of Stack in Recursion (Factorial and Fibonacci).

Queues: Definition, Representation, ADT, Primitive Operations of Queue and its Implementation; Circular Queues using Dynamic Arrays, Priority Queues, Single-Ended and Double-Ended Queues, Multiple Stacks and Queues.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https:// nptel.ac.in/courses/106106133
  https://nptel.ac.in/courses/106106127

Unit III

- **Linked List:** Singly Linked Lists and Chains, Representation of Linked Lists in Memory, Representing Chains in C, Linked List Operations: Traversing, Searching, Insertion, and Deletion, Linked List Stacks and Queues, Polynomial Representation using Linked Lists and Polynomial Addition, Circular Linked Lists, Operations on Circular Linked Lists, Header Linked Lists, Doubly Linked Lists, Operations on Doubly Linked Lists**.**
- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https:// nptel.ac.in/courses/106106133
  https:// nptel.ac.in/courses/106102064
  https:// nptel.ac.in/courses/106106127

Unit IV

Trees: Introduction, Binary Trees, Properties of Binary Trees, Binary Tree Traversals, Binary Search Trees: Definition, Searching, Insertion, Deletion, Joining and Splitting, Height of a Binary Search Tree, AVL Trees: Definition, Rotations, Insertion and Deletion.
Graphs: Introduction, Definition and Graph Representations, Elementary Graph Operations, Traversal Methods: Breadth First Search and Depth First Search.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https:// nptel.ac.in/courses/106106133
  https://nptel.ac.in/courses/106102064
  https://nptel.ac.in/courses/106106127

Unit V

Hashing: Introduction, Hash Table Organizations, Hashing Functions, Static Hashing, Dynamic Hashing, Collision Resolution: Linear Probing, Quadratic Probing, Chaining.
Multiway Search Trees: B-Trees: Definition, Properties, Number of Elements in a B-Tree, Insertion and Deletion. B+ Trees: Definition, Searching, Insertion and Deletion**.**

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https:// nptel.ac.in/courses/106106133
  https ://nptel.ac.in/courses/106102064
  https://nptel.ac.in/courses/106106127

Text Book:

1. Horowitz, E., Sahni, S., and Anderson-Freed, S., Fundamentals of Data Structures in C, 2nd Edition, Universities Press, 2023. ISBN: 9788173716058

Reference Books:

1. Aaron M. Tanenbaum, Yedidyah Langsam, and Moshe J. Augenstein, Data Structures Using C, 2nd Edition, PHI, 2009.
2. Michael J. Folk, Bill Zoellick, and Greg Riccardi, File Structures: An Object-Oriented Approach with C++, Pearson Education, 2004.
3. Seymour Lipschutz, Data Structures: Schaum's Outline Series, Revised 1st Edition, McGraw Hill, 2014.
4. Gilberg and Forouzan, Data Structures: A Pseudocode Approach with C, 2nd Edition, Cengage Learning, 2014.
5. Reema Thareja, Data Structures Using C, 3rd Edition, Oxford University Press, 2012.
6. Jean-Paul Tremblay and Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Edition, McGraw Hill, 2013

Self-Learning: GATE-Oriented Preparation, Assignment Practice, and Complex Program Design

Course Outcomes(COs):

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

1. Apply data structures and dynamic memory management concepts (PO:1, 2, 3, 9, 10; & PSO:1)
2. Analyse linear data structures for problem solving. (PO:1, 2, 3, 9, 10; & PSO:1)
3. Design solutions using linear and non-linear data structures. (PO:1, 2, 3, 9, 10, 11; & PSO:1)
4. Implement tree and graph data structures. (PO :1, 2, 3, 8, 9, 10, 11; PSO: 1)
5. Evaluate advanced data organization techniques. (PO :1, 2, 3, 8, 9, 10, 11; PSO: 1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
GATE -Oriented Aptitude Tests	10	CO1, CO2, CO3, CO4, CO5
Programming Assignment	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

Object Oriented Programming with Java	
Course Code: IS34	**Credits: 3:0:0:3**
Pre – requisites: C Programming	**Contact Hours: 45L+45S**
Course Coordinator: Dr. D. Evangeline	

Course Content

Unit I

Introduction to OOPS using Java: Overview of OOP – Java Buzzwords – Overview of Java – Object oriented programming – Data Types, Variables and Arrays – Operators – Control Statements – Introducing classes in Java – Constructors-Methods -Access specifiers - Static members- Introducing final - Java Doc comments – Overloading Methods

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=18&lesson=19
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=18&lesson=20
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=18&lesson=21

Unit II

A closer look at Methods and Classes – Objects as Parameters – Returning Objects.
Inheritance: Basics– Types of Inheritance -Super keyword -Method Overriding – Dynamic Method Dispatch –Abstract Classes – final with Inheritance.
String Handling: String Constructors – String Length – String Concatenation and Comparison – Searching Strings – Modifying a String – String Buffer Constructors and Methods.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=34&lesson=37
  https://www.youtube.com/watch?v=GnpEm89Y0fg&list=PLd3UqWTnYXOlrdBwt2DTnF3u3W-1Kozk0

Unit III

Packages and Interfaces: Packages – Packages and Member Access –Importing Packages – Interfaces.
Exception Handling: Fundamentals – Multiple catch Clauses – Nested try Statements – Java's Built-in Exceptions – Creating your own Exception Subclasses.
Multithreaded Programming: Java Thread Model–Creating a Thread and Multiple Threads – Priorities.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:**
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=42&lesson=44
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=50&lesson=53
  https://onlinecourses.nptel.ac.in/noc25_cs110/unit?unit=58&lesson=60

Unit IV

Java Collections: Overview, Collection Interfaces – List, Set, Collection Classes – ArrayList, LinkedList, Accessing a Collection via an Iterator, **Working with Maps**: Map.Entry Interface – HashMap Class – Comparators. Exceptions in Collection Classes.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:** https://www.youtube.com/watch?v=rH0winlka8A

Unit V

Wrappers: Auto boxing – Unboxing. **Generics:** Generic classes – Generic Methods – Bounded Types – Generic Interfaces - Restrictions and Limitations. **Lambda Expressions:** Fundamentals, Functional Interfaces

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and Programming Demonstration.
- **Links:** https://www.youtube.com/watch?v=K1iu1kXkVoA

Text Book:

1. Herbert Schildt, "Java: The Complete Reference", 11th Edition, McGraw Hill Education, New Delhi, 2019.
2. E. Balagurusamy, "Programming with Java", 7th Edition, McGraw Hill Education, New Delhi, 2024.

Reference Book:

1. Cay S. Horstmann, "Core Java Fundamentals", Volume 1, 11th Edition, Prentice Hall, 2018.

Self-Learning: Programming Assignment**,** Mini project, MCQ.

Course Outcomes(COs):

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

1. Apply the concepts of classes and objects to solve simple problems. (PO :1, 2, 3; PSO: 1)
2. Develop programs using objects as arguments or return type, inheritance and string handling. (PO :1, 2, 3, 4; PSO: 1)
3. Apply the concepts of packages, interfaces, exception handling and multithreaded model to solve real world problems. (PO :1, 2, 3, 4; PSO: 1)
4. Build efficient and reusable Java applications with relevant classes of Java Collection Framework. (PO :1, 2, 3, 4, 6; PSO: 1)
5. Design and implement scalable Java applications with Wrappers, Generics and Lambda Expressions. (PO :1, 2, 3, 4, 6 ; PSO: 1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Mini Project and Programming Assignment	20	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

Discrete Mathematical Structures	
Course Code: IS35	**Credits:** 3:0:0:3
Pre – requisites: NIL	**Contact Hours:** 45L + 45S
Course Coordinator: Prashanth Kambli	

Course Content

Unit I

Introduction to Discrete Mathematics: Logic, Propositions and Logical Operations, Conditional statements, Methods of proof.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk
- **Links:** https://nptel.ac.in/courses/106106094
- **Impartus recording:** https://a.impartus.com/ilc/#/course/107790/533

Unit II

Relations and Digraphs: Relations and digraphs, Paths in relations and digraphs, Properties of relations, Equivalence relations, Operations on relations, Transitive closure and Warshall's algorithm.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk
- **Links:** https://nptel.ac.in/courses/106106094
- **Impartus recording:** https://a.impartus.com/ilc/#/course/107790/533

Unit III

Functions: Types of Functions, Functions for computer Science, Permutation functions, POSETS: Order relations and structures: Extremal elements of partially ordered sets, Lattices.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk
- **Links:** https://nptel.ac.in/courses/106106094
- **Impartus recording:** https://a.impartus.com/ilc/#/course/107790/533

Unit IV

Graphs: Graph terminology and special types of graphs, Representing graphs and graph isomorphism, connectivity, Euler and Hamilton paths.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk
- **Links:** https://nptel.ac.in/courses/106106094
- **Impartus recording:** https://a.impartus.com/ilc/#/course/107790/533

Unit V

Semigroups and Groups: Binary operations revisited, Groups, Semigroups: Sub semigroup, Sub monoid, Isomorphism, Homomorphism.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk
- **Links:** https://nptel.ac.in/courses/106106094
- **Impartus recording:** https://a.impartus.com/ilc/#/course/107790/533

Text Books:

1. Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, Discrete Mathematical Structures, 6th edition, PHI (all topics except graphs).

2. Kenneth H Rosen, Discrete Mathematics and its applications, 6th Edition, Tata McGraw-Hill.

Reference Books:

1. Ralph P.Grimaldi, B.V Ramana, Discrete and Combinatorial Mathematics, Fifth edition.
2. J.P.Trembly, R. Manohar, Discrete mathematical structures with applications to Computer Science, McGraw Hill.

Self-Learning: Self-Learning: GATE oriented problem solving, Study real-world applications of propositional logic, explore applications of relations, investigate the role of functions and lattices, Study graph applications in GPS navigation, computer networks, social networks, and web search engines, explore the application of algebraic
structures.

Course Outcomes(COs):

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

1. Apply the concepts of logical reasoning to verify the correctness of mathematical statements (PO:1, 2, 9, 10, 12; PSO:1)
2. Analyze the properties and operations of relations in solving various problems. (PO:1, 2, 9, 10, 12; PSO:1)
3. Understanding the concepts of functions and partially ordered sets for solving the given problem. (PO:1, 2, 9, 10, 12; PSO:1)
4. Apply the graph theory concepts in solving computational problems. (PO:1, 2, 9, 10, 12; PSO:1)
5. Apply the concepts of groups for binary operations. (PO:1, 2, 9, 10, 12; PSO:1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Problem Solving & Quiz	10	CO1, CO2, CO3, CO4,CO5
Case Study & Seminar	10	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

Universal Human Values	
Course Code: UHV36	**Credits: 2:0:0:2**
Pre – requisites: NIL	**Contact Hours: 30L+30S**
Course Coordinator: Mr. Mushtaq Ahmed D M	

Course Content

Unit I

Need, Basic Guidelines, Content and Process for ValueEducation

1. Understanding the need, basic guidelines, content and process for Value Education
2. Self-Exploration–what is it? - its content and process; 'Natural Acceptance' and Experiential Validation- as the mechanism for self-exploration
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facilities- the basic requirements for fulfillment of aspirations of every human being with their correct priority
5. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario
6. Method to fulfill the above human aspirations: understanding and living in harmony at various levels

- **Pedagogy / Course delivery tools:** Chalk and talk, Power point presentation, Videos.
- **Lab component / Practical Topics**: Survey/polls for self-exploration
- **Links**: Holistic Development and Role of Education https://youtu.be/sGZtTPe- lhQ

Unit II

Understanding Harmony in the Human Being - Harmony in Myself!

1. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
2. Understanding the needs of Self ('I') and 'Body' - Sukh and Suvidha
3. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
4. Understanding the characteristics and activities of 'I' and harmony in 'I'
5. Understanding the harmony of I with the Body: Sanyam and Swasthya; correct appraisal of Physical needs, meaning of Prosperity in detail
6. Programs to ensure *Sanyam* and *Swasthya*

Practice Exercises and Case Studies will be taken up in Practice Sessions.

- **Pedagogy / Course delivery tools:** Chalk and talk, Power point presentation, Videos.
- **Lab component / Practical Topics:** Survey/polls for self-exploration
- **Links**: Harmony in Human Being- Self and Body https://youtu.be/0ERSMkRPQBM
- **Links**: Harmony in Human Being- Self https://youtu.be/83oGJ4oDeIg
- **Links**: Harmony between Self and Body Prosperity https://youtu.be/aJ_BU2OgpKs

Unit III

Understanding Harmony in the Family and Society- Harmony in Human- Human Relationship

1. Understanding Harmony in the family – the basic unit of human interaction
2. Understanding values in human-human relationship; meaning of Nyaya and program for its fulfillment to ensure Ubhay-tripti;
3. Trust (Vishwas) and Respect (Samman) as the foundational values of relationship
4. Understanding the meaning of Vishwas; Difference between intention and competence
5. Understanding the meaning of Samman, Difference between respect and differentiation; the other

salient values in relationship

6. Understanding the harmony in the society (society being an extension of family):
7. Samadhan, Samridhi, Abhay, Sah-astitva as comprehensive Human Goals
8. Visualizing a universal harmonious order in society- Undivided Society (Akhand Samaj), Universal Order (Sarvabhaum Vyawastha)- from family to world family!

Practice Exercises and Case Studies will be taken up in Practice Sessions.

- **Pedagogy / Course delivery tools:** Chalk and talk, Power point presentation, Videos.
- **Lab component / Practical Topics**: Survey/polls for self-exploration
- **Links**: Harmony in Family- Trust https://youtu.be/F2KVW4WNnS8
- **Links**: Harmony in family- Respecthttps://youtu.be/iLqNRPuv0_8
- **Links**: Harmony in family- Other Feeling Justice https://youtu.be/TcYJB7reKnM
- **Links**: Harmony in the Society https://youtu.be/BkWgFinrnPw

Unit IV

Understanding Harmony in the Nature and Existence - Whole existence as Co- existence

1. Understanding the harmony in the Nature
2. Interconnectedness and mutual fulfillment among the four orders of nature- recyclability and self-regulation in nature
3. Understanding Existence as Co-existence (Sah-astitva) of mutually interacting units in all-pervasive space
4. Holistic perception of harmony at all levels of existence

Practice Exercises and Case Studies will be taken up in Practice Sessions.

- **Pedagogy / Course delivery tools**: Chalk and talk, Power point presentation, Videos.
- **Lab component / Practical Topics**: Survey/polls for self-exploration
- **Links**: Harmony in Nature https://youtu.be/K1Jpd_ojydw
- **Links**: Harmony in Existence https://youtu.be/mormUeZ_RUE

Unit V

Implications of the above Holistic Understanding of Harmony on Professional Ethics

1. Natural acceptance of human values
2. Definitiveness of Ethical Human Conduct
3. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
4. Competence in professional ethics:
   a. Ability to utilize the professional competence for augmenting universal human order
   b. Ability to identify the scope and characteristics of people-friendly and eco- friendly production systems,
   c. Ability to identify and develop appropriate technologies and management patterns for above production systems.
5. Case studies of typical holistic technologies, management models and production systems
6. Strategy for transition from the present state to Universal Human Order:
   a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers
   b. At the level of society: as mutually enriching institutions and organizations

Suggested Learning Resources:

- **Pedagogy / Course delivery tools**: Chalk and talk, Power point presentation, Videos.
- **Lab component / Practical Topics**: Survey/polls for self-exploration

Text Books:

1. **R.R Gaur, R Sangal, G P Bagaria**, A foundation course in Human Values and professional Ethics, Excel books, New Delhi, 2010, ISBN 978-8-174-46781-2

Reference Books:

1. B L Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co., Lucknow. Reprinted 2008.
2. PL Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
3. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
4. Ivan Illich, 1974, Energy & Equity, The Trinity Press, Worcester, and HarperCollins, USA
5. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, limits to Growth, Club of Rome's Report, Universe Books.
6. Subhas Palekar, 2000, How to practice Natural Farming, Pracheen(Vaidik) Krishi Tantra Shodh, Amravati.
7. A Nagraj, 1998, Jeevan Vidya ek Parichay, Divya Path Sansthan, Amarkantak.
8. E.F. Schumacher, 1973, Small is Beautiful: a study of economics as if people mattered, Blond & Briggs, Britain.
9. A.**N. Tripathy**, 2003, Human Values, New Age International Publishers.

Web links and Video Lectures (e-Resources):

1. https://www.youtube.com/channel/UCQxWr5QB_eZUnwxSwxXEkQw
2. https://www.youtube.com/watch?v=P4vjfE-YnVk&list=PLWDeKF97v9SP7wSlapZcQRrT7OH0ZlGC4
3. Course handouts: https://drive.google.com/drive/folders/1zioX_4L2fCNX4Agw282PN86pcZZT3Osr?usp=sharing
4. Presentation slides: https://drive.google.com/drive/folders/1rMUKh1s0HPRBlpp_b1mpS-duNRcwS6YH?usp=sharing

Course Outcomes (COs):

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

1. Apprehend the need of Value Education over Human aspirations (PO: 6)
2. Assimilate Harmony over the physical needs and to overcome the self- needs for a prosperous life. (PO: 6)
3. Recognize the need of Harmony in the Family and Society for a better World. (PO: 6)
4. Explain the need of mutual understanding for Holistic Harmony in all the Levels of Human Existence. (PO: 6)
5. Explain the Holistic understanding of Harmony and Professional Ethics at Individual Level and Society. (PO: 6, 8)

DATA STRUCTURES LAB	
Course Code: ISL38	**Credits: 0:0:1**
Pre – requisites: Fundamentals of Computing	**Contact Hours: 15P**
Course Coordinator: Dr. Kusuma S	

Course Content

1. Write a C program to store and manage a list of records (e.g., patients, students, or employees) using structures, pointers, and dynamic memory allocation. Allow the user to add, view, and search records. Use malloc(), realloc(), and free() for memory allocation and deallocation.
2. Design, develop, and implement a menu-driven program to perform Stack operations using an array, including Push, Pop, Display, Overflow/Underflow handling, and an application to check whether a string is a palindrome.
3. Write a program to convert and print a given valid parenthesised infix arithmetic expression into a postfix expression. The expression consists of single-character operands and binary operators (+, –, *, /). Apply the concept of the Stack data structure to solve this problem.
4. Write a program to evaluate a valid postfix expression using a Stack. Assume that the postfix expression is read as a single line consisting of non-negative single-digit operands and binary operators. The operators are (+, –, *, /).
5. A call centre phone system has to maintain customer calls and provide service based on their arrival order. Write a C program to simulate this system using a Queue data structure. The program should provide options to add and remove phone calls in the appropriate order for service.
6. Implement a keyboard input buffer using a Circular Queue (array implementation) with operations for insertion, deletion, display, and status checking (full/empty). Demonstrate efficient reuse of freed buffer locations.
7. Write a program to illustrate memory allocation to files based on their size using a Singly Linked List. The program must support the following operations:
    a) Insert a node
    b) Delete a node
    c) Display the list
8. Design, Develop, and Implement a Program in C for the following operations on a Singly Linked List:
    a) Represent and evaluate a polynomial $P(x) = 6x^2 + 3x^3 + 2x$
    b) Find the sum of two polynomials POLY1(x) and POLY2(x) and store the result in POLYSUM(x).
    c) Support the program with appropriate functions for each of the above operations.
9. Write a program to illustrate memory allocation to files based on their size using a Doubly Linked List. The program must support the following operations:
    a. Insert a node
    b. Delete a node
    c. Display the list
10. Design, Develop, and Implement a menu-driven program to perform Undo operations in a text editor using a Stack implemented with a Singly Linked List. The program should support storing actions, undoing recent actions, displaying stored actions, and handling empty-stack conditions.
11. Design, Develop, and Implement a menu-driven program to simulate a printer queue using a Queue implemented with a Singly Linked List. The program should support the following operations:
    a) Add a print job
    b) Process the next print job

c) Display pending print jobs
d) Check queue status
e) Exit

Demonstrate how the linked-list implementation supports dynamic job allocation without queue size restrictions.

12. A hospital database consists of CT images of patients. To facilitate efficient retrieval of image records, implement database indexing using a Binary Search Tree (BST). Store image identification numbers in the BST and perform preorder traversal to display the indexed records.

Text Book:

1. Reema Thareja, Data Structures Using C, 2nd Edition, Oxford University Press, 2014. ISBN: 9780198099307.
2. Yashavant Kanetkar, Data Structures Through C, 4th Edition (Revised & Updated), BPB Publications, 2022. ISBN: 9789355511898.

Reference Books:

1. M. Tanenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C", 2nd Edition, PHI, 2009.
2. Michael J. Folk, Bill Zoellick and Greg Riccardi, "File Structures-An Object Oriented Approach with C++", Pearson Education, 2004
3. Semour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
4. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
5. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw-Hill, 2013

Course Outcomes(COs):

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

1. Design the experiment for the given problem using Data Structures concepts. (PO: 1, 2, 3, 8, 9, 1; PSO: 1,3)
2. Develop the solution for the given real-world problem. (PO: 1, 2, 3, 4, 5, 8, 9; PSO:1)
3. Analyze the results and produce substantial written documentation (PO:1, 2, 4, 5, 8, 9; PSO:1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course Outcomes Attained**
Internal Test	20	CO1, CO2, CO3
Other Components		
Weekly Evaluation- Lab Record	20	CO1, CO2, CO3
Project	10	CO1, CO2, CO3
The Final CIE out of 50 Marks = Marks Scored in Lab Test + Marks of Lab Record + Marks Scored in Quiz		
Semester-End Examination (SEE)	50	CO1, CO2, CO3

Object Oriented Programming with Java Lab	
Course Code: ISL39	**Credits: 0:0:1**
Pre – requisites: C Programming Lab	**Contact Hours: 15L**
Course Coordinator: Mrs. Prathima	

Course Content

1. **Classes and Objects:** Write a Java program to create a Student class with data members studentId, studentName, and marks obtained in five subjects. Read all student details and marks from the user at runtime. Create an array of five Student objects and initialize the data members without using constructors. Calculate the total marks obtained by each student and identify the student who has secured the maximum total marks and the student who has secured the minimum total marks. Finally, display the details of all students along with their total marks, and print the details of the students with the highest and lowest total marks.
2. **Constructors and Objects as Arguments / Return Type:** Write a Java program to create a Product class with data members productId, productName, category, and price. Initialize the data members using user input using constructors. Override the toString() method to return the product details as a formatted string. Define a method displayProduct(Product p) that accepts a Product object as an argument and displays its details, and another method getProduct() that returns a Product object. Create a Product object, invoke displayProduct() by passing the object as an argument, obtain the object using getProduct(), and display its details using the overridden toString() method.
3. **Use of super and abstract keywords:** Create an abstract class Payment with the data members transactionId and amount. Initialize these data members using a constructor. Declare an abstract method processPayment() and define a concrete method displayTransaction() to display the transaction details. Create three subclasses, namely CreditCardPayment, UPIBasedPayment, and NetBankingPayment, that extend the Payment class and provide implementations for the processPayment() method. Create separate objects of all three payment types and invoke the processPayment() and displayTransaction() methods to display the transaction details and payment processing information.
4. **Method overriding:** Create a superclass FoodOrder with the data members orderId, customerName, and orderAmount. Define a method calculateBill(). Create subclasses DineInOrder, TakeAwayOrder, and OnlineOrder that extend the FoodOrder class. Each subclass should implement the calculateBill() method by applying different service charges or discounts. Create objects of all three order types and invoke the calculateBill() method using a FoodOrder reference.
5. **Static data members and static methods:** Create a class Course with the data members courseId, courseName, and credits. Declare a static data member total_num_stud_Enrolled that represents the total number of students enrolled and is common to all Course objects. Initialize the course details using a parameterized constructor and update the static data member whenever a Course object is created. Define a static method displayTotalEnrollment() to display the total number of students enrolled. Create multiple Course objects and display their details along with the total number of students enrolled.
6. **Array of objects and Method Overloading:** Write a Java program to create an ElectricityConsumer class with the data members consumerId, consumerName,

unitsConsumed, and ratePerUnit. Initialize the data members using a parameterized constructor. Overload the method calculateBill() such that one method calculates the electricity bill as the product of unitsConsumed and ratePerUnit, while the other method accepts an additional surcharge percentage as an argument and calculates the revised bill using the formula Revised Bill = Bill + (Bill × surchargePercentage / 100). Create an array of ElectricityConsumer objects and accept the details of multiple consumers from the user. Display the consumer details, the original electricity bill, and the revised bill after applying the surcharge by invoking both overloaded methods.

7. **String and StringBuffer:** Write a Java program to create a class StringOperations that accepts a string from the user. Using the String class, implement the methods
   - stringLength() – Returns the length of the string.
   - extractSubstring(int start, int end) – Returns a substring.
   - replaceCharacter(char oldChar, char newChar) – Replaces all occurrences of a character.

   Using the StringBuffer class, implement the methods
   - reverseString() to reverse the original string.
   - findFirstOccurrence(char ch) to determine the index of the first occurrence of a specified character.
   - findLastOccurrence(char ch) to determine the index of the last occurrence of the specified character in the original string.

Implement a menu-driven program that displays the result of the selected operation.

8. **Packages and Interfaces:** Define a package named stringoperations to encapsulate the string processing functionality. Create an interface named StringManipulator with at least eight methods for string operations:
   - String reverse(String input): Reverses the given string.
   - String toUpperCase(String input): Converts the string to uppercase.
   - String toLowerCase(String input): Converts the string to lowercase.
   - String concatenate(String str1, String str2): Concatenates two strings.
   - int countVowels(String input): Counts the number of vowels in the string.
   - int wordCount(String input): Counts the number of words in the string.

   Implement the interface in a class named StringProcessor within the stringoperations package, providing concrete implementations for all methods. Create a main class in a separate package application, where an object of StringProcessor is created. The user is prompted to enter a string. Each method of the interface is tested and its output is displayed using a menu-driven program.

9. **Exception Handling:** Create a class Queue with the following methods:
   - enqueue(int element) to insert an element into the queue.
   - dequeue() to remove an element from the queue.
   - display() to display all the elements present in the queue.

Define two user-defined exceptions, QueueOverflowException and QueueUnderflowException, to handle a full queue and an empty queue respectively. Implement a menu-driven program to perform the queue operations and handle exceptions by displaying suitable error messages whenever an exception occurs.

10. **Java Collections (ArrayList and HashMap):** Write a menu driven Java program to perform the following operations:

- Create a book database and store it in a Collection List. Book class should include title, author, publisher and price data members.
- Using a HashMap, maintain the book details with respect to a unique book id.
- Write a method to sort the books in ascending order of price and store it in another List.
- Prompt for an author name and list all the books with the same author name.
- Create a new list holding all the book details with price greater than a user specified price.

11. **Java Collections (LinkedList):** Create a helper class MissedCall with the data members callerNumber, callerName, and callTime.
    - If the caller's name is unavailable, assign the value "Private Caller".
    - The system should maintain the missed calls in the order they are received and store a maximum of three missed calls at any time.
    - When a new missed call is received after the queue is full, automatically remove the oldest missed call and insert the new one.
    - Delete a selected missed call from the collection.

    Implement a menu-driven program to display the updated list of missed calls after each operation.
12. **Generics and Lambda Expression:** Create generic Stack class that performs the following operations:
    - push(T element) to insert element on top of Stack.
    - pop() to remove and return element on top of Stack.
    - isEmpty() that displays true if the Stack is empty.
    - display() that displays the elements in Stack. (Use Lambda expressions to display elements in a Stack)
    - search(T element) is a generic method that searches for a given element and displays its position.

    Demonstrate creating Stack of String and Integer objects using a menu-driven Java program.

Text Book:

1. Herbert Schildt, “Java: The Complete Reference”, 11 th Edition, McGraw Hill Education, New Delhi, 2019.
2. E. Balagurusamy, “Programming with Java”, 7th Edition, McGraw Hill Education, New Delhi, 2024.

Reference Book:

1. Cay S. Horstmann, “Core Java Fundamentals”, Volume 1, 11 th Edition, Prentice Hall, 2018.

Course Outcomes(COs):

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

1. Apply concepts such as Classes and Objects, Constructors, Inheritance and Overriding to develop real-world Java applications. (PO :1, 2, 3, 5, 8, 9; PSO: 1)
2. Design and implement programs with static data members / methods; array of objects and method overloading; String and StringBuffer classes; and Interfaces and Packages for data processing and management tasks. (PO :1, 2, 3, 4, 5, 8, 9; PSO: 1)
3. Develop Java applications using Exception Handling, Java Collections, Generics and Lambda Expressions for solving real-world problems (PO :1, 2, 3, 5, 6, 8, 9; PSO: 1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course Outcomes Attained**
Internal Test	20	CO1, CO2, CO3
Other Components		
Weekly Evaluation- Lab Record MiniProject and Viva	30	CO1, CO2, CO3
The Final CIE out of 50 Marks = Marks Scored in Lab Test + Marks of Lab Record + Marks Scored in MiniProject and Viva		
Semester-End Examination (SEE)	50	CO1, CO2, CO3

Introduction to Web Programming	
Course Code: ISAEC371	**Credits: 1:0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L+15S**
Course Coordinator: Lincy Meera Mathews	

Course Content

Unit I

Introduction to Web Programming: Structural Elements, Title Element, meta-Element, HTML Attributes, Body Element, Block Elements, block quote Element, Whitespace Collapsing, pre-Element, Phrasing Elements, Editing Elements, q and cite Elements, dfn, abbr, and time Elements, br and wbr Element, sub, sup, s, mark, and small Elements, strong, em, b, u, and i Elements, Span Element

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Demonstration of code
- **Links:** https://www.w3schools.com/html/html_attributes.asp

Text Book1: Chapter 1(1.5,1.6,1.7,1.8,1.9,1.13) and

Unit II

HTML5: Hello HTML5, Loose Syntax Returns, XHTML5, HTML5: Embracing the Reality of Web Markup, Presentational Markup Removed and Redefined, HTML5 Document Structure Changes, Adding Semantics: Marking Text, Indicating Dates and Time, Inserting Figures, HTML5's Open Media Effort: Audio, Video, Client-Side Graphics with: Drawing and styling lines and shapes.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Demonstration of code
- **Links:** https://www.w3schools.com/charsets/tryit.asp?deci=193&ent=Aacute

Unit III

Introduction to CSS: CSS Overview, CSS Rules, Example with Type Selectors and the Universal Selector, CSS Syntax and Style, Class Selectors, ID Selectors, span and div Elements, Cascading, style Attribute, style Container, External CSS Files, CSS Properties, Color Properties, RGB Values for Color, Opacity Values for Color, HSL and HSLA Values for Color, Font Properties, line-height Property, Text Properties, Border Properties, Element Box, padding Property, margin Property.

- **Pedagogy/Course delivery tools:** Chalk and talk, PowerPoint Presentation, Demonstration of code
- **Links: https://www.w3schools.com/css/css_syntax.asp**

Unit IV

Tables and CSS, Links and Images: Table Elements, formatting a Data Table: Borders, Alignment, and Padding, CSS Structural Pseudo Class Selectors, thead and tbody Elements, Cell Spanning, CSS display Property with Table Values, a Element, Different Types of href Values, Relative URLs, CSS for Links, Bitmap Image Formats: GIF, JPEG, PNG, img Element, Positioning Images, Shortcut Icon, iframe Element.Backgroun images, web fonts

- **Pedagogy/Course delivery tools:** Chalk and talk, PowerPoint Presentation, Demonstration of code
- **Links**:https://www.w3schools.com/css/css_table.asp, https://www.w3schools.com/css/css_link.asp

Unit V

Introduction to JavaScript: History of JavaScript, Hello World Web Page, Buttons, Functions, Variables, Identifiers, Assignment Statements and Objects, Document Object Model, Forms, Form Element, Controls, Text Control, Email address Generator Web Page, Event-Handler Attributes. Pedagogy/Course delivery tools: Chalk and talk, Power Point Presentation, Demonstration of code

- **Links**: https://www.w3schools.com/js/js_functions.asp

Text Books:

1. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, First Edition
2. HTML & CSS: The Complete Reference Thomas A. Powell, Fifth Edition, Tata McGraw Hill

Reference Books:

1. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 1st Edition, Pearson Education India. (ISBN:978-9332575271)
2. David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", 1st Edition, O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014 (ISBN:978- 9351108078)
3. Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016. (ISBN:978-9352133246)

Self-Learning: Design a front end using HTML, CSS and Javascript.

Course Outcomes (COs):

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

1. Use different HTML tags for web page design. (PO: 1,3)
2. Develop web pages using HTML5 constructs (PO: 1, 3)
3. Design web pages using basic features of CSS and HTML (PO:1, 2, 3)
4. Apply advanced features of CSS and HTML to create web pages. (PO: 1,2,3) CO5.
5. Apply features of JavaScript to handle client-side interactions. (PO:1, 2, 3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes addressed**
Internal test-I	30	CO1, CO2
Internal test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other components		
Mini Project	20	CO1, CO2, CO3, CO5, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

Green IT and Sustainability	
Course Code: ISAEC372	**Credits: 1:0:0:1**
Pre – requisites: Nil	**Contact Hours: 15L+15S**
Course Coordinator: Dr. Rajeshwari S B	

Course Content

Unit I

Introduction: Foundations of sustainable design and HCI, sustainability and usability concerns, scope and objectives of sustainable interaction design.

- **Pedagogy/Course Delivery tools:** Power Point Presentation
- **Links:** https://nptel.ac.in/courses/106101060

Unit II

Usability and Human–Computer Interaction (HCI): Usability principles, evaluation of usability, role of HCI in sustainable design, challenges in achieving usable and sustainable systems

- **Pedagogy/Course Delivery tools:** Power Point Presentation
- **Links:** https://nptel.ac.in/courses/107103081

Unit III

User Participation in the System Development Process: Importance of user involvement, participatory design, sustainability through user-centered design approaches.

- **Pedagogy/Course Delivery tools:** Power Point Presentation
- **Links:** https://nptel.ac.in/courses/107103081

Unit IV

Physical, Cognitive and Affective Engineering; Color, Prototyping and Navigation; Principles and Guidelines, Design Evaluation and Testing; Task Analysis.

- **Pedagogy/Course Delivery tools:** Power Point Presentation
- **Links:** https://nptel.ac.in/courses/107103081

Unit V

Models and Methodologies; New Participative Methodology for Sustainable Design (NPMSD); Future ICTs—Present Trends for Future Developments.

- **Pedagogy/Course Delivery tools:** Power Point Presentation
- **Links:** https://nptel.ac.in/courses/107103081

Text Book:

1. Sustainable Design: HCI, Usability and Environmental Concerns, Tomayess Issa, Pedro Isaias, 2nd Edition, Springer, 2022.

Reference Book:

1. Green Information Technology – A Sustainable Approach, Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Elsevier, 2015.

Self-Learning:

Analyze and Present a case study on Green IT and sustainability initiatives adopted by various industries and organizations.

Course Outcomes(COs):

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

1. Comprehend the foundational concepts of Green IT and its importance in addressing global environmental challenges. (PO:2,6,8,9; PSO:2).
2. Analyze the environmental impact of IT infrastructure, including data centers, and propose strategies for energy efficiency and resource optimization. (PO:2,6,8,9; PSO:2).
3. Evaluate sustainable computing practices to minimize the carbon footprint of IT operations. (PO:2,6,8,9; PSO:2).
4. Develop and implement policies for responsible e-waste management and sustainable IT procurement. (PO:2,6,8,9; PSO:2).
5. Apply sustainable software development principles to create energy-efficient and eco-friendly applications. (PO:2,6,8,9; PSO:2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Case Study Presentation	10	CO1, CO2, CO3, CO4,CO5
MCQ Test	10	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

Engineering Economics	
Course Code: ISAEC373	**Credits: 1:0:0:1**
Pre – requisites: NIL	**Contact Hours:15L + 15S**
Course Coordinator: Dr Jagadeesh Sai D	

Course Contents

Unit 1

Economics: Definition and Scope of Engineering Economics, Flow in an Economy, Law of Supply and Demand, Concept of Engineering Economics, Types of Efficiency, Elements of Costs, Other Costs/Revenues, Break-Even Analysis, Profit/Volume Ratio (P/V Ratio), Elementary Economic Analysis, Design Selection for A Product, Building Material Selection, Process Planning/Process Modification

- **Pedagogy**: Lecture, chalk-and-board explanation, numerical problem solving, real-life examples
- **Links**: https://nptel.ac.in/courses/112107209

Unit 2

Interest Formulas and Their Applications: Time Value of Money, Interest Formulas, Single-Payment Compound Amount, Present Worth Amount, Equal-Payment Series Compound Amount, Equal-Payment Series Sinking Fund, Present Worth Method of Comparison, Revenue-Dominated Cash Flow Diagram, Cost-Dominated Cash Flow Diagram, Future Worth Method.

- **Pedagogy**: Lecture, worked examples, case-based discussion, tutorial problems
- **Links**: https://nptel.ac.in/courses/112107209

Unit 3

Depreciation: Methods of Depreciation, Straight Line Method of Depreciation, Declining Balance Method of Depreciation, Sum-Of-The-Years-Digits Method of Depreciation. Evaluation of Public Alternatives, Inflation Adjusted Decisions, Procedure to Adjust Inflation, Inflation Adjusted Economic Life of Machine

- **Pedagogy**: Lecture, numerical demonstrations, tutorial sheets, problem-solving sessions
- **Links**: https://nptel.ac.in/courses/112107209

Unit 4

Inventory Control: Purchase Model, Manufacturing Model, Make or Buy Decision, Approaches for Make or Buy Decision, Simple Cost Analysis, Economic Analysis, Break-Even Analysis

- **Pedagogy**: Lecture, numerical problem solving, exam-oriented practice
- **Links**: https://nptel.ac.in/courses/112107209

Unit 5

Project Management: Phases of Project Management, Guidelines for Network Construction, Critical Path Method (CPM), Gantt Chart/Time Chart, Pert (Project Evaluation and Review Technique), Value Analysis/Value Engineering 2 When to Apply Value Analysis, Value Engineering Procedure

- **Pedagogy**: Lecture, numerical problems, case studies, revision discussions
- **Links**: https://nptel.ac.in/courses/112107209

Text Books:

1. Engineering Economics, R. Panneerselvam**,** PHI Learning Private Limited**,** 2nd Edition, 2012**.**
2. Engineering Economy, William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling**,** Pearson Education**,** 17th Edition, 2022**.**

Reference Book:

1. Prasanna Chandra, Fundamentals of Financial Management, McGraw-Hill 2.

Self-Learning Case Study

Course Outcomes:

After completing the course, the student will be able to:

1. Understand basic economic principles relevant to engineering decisions (PO:1, 2, 11; PSO: 1)
2. Analyze cost structures and break-even conditions (PO:1, 2, 3, 4, 11; PSO:1)
3. Apply time value of money concepts (PO:1, 2, 3, 11; PSO:1)
4. Evaluate engineering projects using economic appraisal techniques (PO:1, 2, 3, 4, 11; PSO:2)
5. Assess replacement, depreciation and inflation effects (PO:1, 2, 3, 4, 11; PSO:1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
MCQ	20	CO1, CO2, CO3, CO4,CO5
Semester-End Examination	100	CO1, CO2, CO3, CO4, CO5

.

Microcontrollers	
Course Code: IS41	**Credits: 3:0:1:3**
Prerequisites: Computer Organization and Architecture	**Contact Hours: 45L + 15P+45S**
Course Coordinator: Mr. Prashanth Kambli	

Course Contents

Unit 1

ARM Embedded Systems, RISC design philosophy, ARM design philosophy, Introduction to Embedded systems, Design of Embedded Systems, Applications, Embedded processors, Operating System, Connectivity. Internal Components of System-on-chip, General Microprocessor unit, MCU PIN diagram, Timers and Counters, Pulse width modulator.

- **Pedagogy**: Lecture, numerical problem solving
- **Links:** Introduction to Embedded System Design - Course
- **Impartus recording:** : https://a.impartus.com/ilc/#/video/id/3404203

Unit 2

Serial Communication, Direct Memory Access, Semiconductor Memory, Designing Low power systems, BUS architecture. Embedded Systems – the Software, Endian-ness, Data Alignment and Memory Banks, Peripheral I/O and Memory Mapped I/O, Load Store Architecture, Stack, FLAGS

- **Pedagogy**: Lecture, numerical problem solving
- **Links:** DMA Introduction: Master – Slave Communication | Free Video Tutorial | Udemy
- **Impartus recording:** : https://a.impartus.com/ilc/#/video/id/3510844

Unit 3

The Architecture of ARM 7, ARM 7 architecture, Interrupts and Exceptions, ARM7 pipeline, Advanced Features. Assembly Programming of ARM7, Embedded program development, ARM7 Instruction set.

- **Links**:https://www.udemy.com/course/armv8-exception-level1/
- **Impartus recording**: https://a.impartus.com/ilc/#/video/id/3542117

Unit 4

Assembly Language Programming, Accessing Memory, Programming of ARM7 using C, ARM7 SOC.

- **Pedagogy**: Lecture, numerical problem solving
- **Links**: Microcontroller Embedded C Programming: Absolute Beginners | Udemy
- **Impartus recording** https://a.impartus.com/ilc/#/course/1259590/1112

Unit 5

Architecture of ARM Cortex-M, Cortex-M Processors, Cortex-M0, Modes and States, Programming Model, Memory Model, Nested Vector Interrupt Controller, Power management using sleep modes.

- **Pedagogy**: Lecture, numerical problem solving

- **Links**: Microprocessors and Microcontrollers: Arm Cortex-M0+ (using RP2040) – NPTEL+
- **Impartus recording** https://a.impartus.com/ilc/#/course/1259590/1112

MICROCONTROLLER LAB

Part A

1. a. ALP to add first 10 odd numbers. Store sum in register.

   b. ALP to compute sum of squares of 5 numbers starting from 1. Write and use procedure SQU. Store sum in register.

2. a. ALP to add the first n even numbers. Store the result in a memory location.

   b. ALP to generate a geometric progression with a limit n. Display the results in memory.

3. a. ALP to count the number of zeroes and ones in a binary number.

   b. ALP to find the average of ten 16-bit numbers stored in memory.

4. a. ALP to find the factorial of a number.

   b. ALP to generate the first n Fibonacci numbers.

5. ALP to find the sum of digits of a number.
6. ALP to convert BCD number to binary.
7. ALP to select a set of r objects from a set of n objects without considering the order of elements in a selection using combination method.
8. ALP to select a set of r objects from a set of n objects considering the order of elements in an arrangement using permutation method.
9. ALP to implement Bubble Sort on an array of integers.
10. ALP to implement Binary Search on an array of integers.
11. ALP to check whether the given number is palindrome.
12. ALP to count the number of times a substring is repeated in the string.

Part B

1. C program to toggle the lowest pin of Port 0 with a delay between the two states. Observe and record the waveform obtained using the Logic Analyzer in the Keil simulator.
2. C program to generate an asymmetric square wave of 120Hz and having a duty cycle of 25% using the Timer0 module.
3. C program to generate a square wave using Timer0 in the interrupt mode.
4. C program to make a LED glow at different brightness levels (low to high) with brightness levels varying over duration of 2s. Demonstrate using logic analyzer window.
5. C program to display the string ‘I LOVE ISE’ in the serial window of UART1
6. Write a C program to Interface NuMicro MCU Learning Board to Light a RGB LED connected to port A12-14.
7. Write a C program to Interface NuMicro MCU Learning Board to beep a buzzer connected to

port B11.

8. Write a C program to Interface NuMicro MCU Learning Board to a even segment to display the values from 0 to 9999.
9. Write a C program to Interface NuMicro MCU Learning Board to a 3x3 keypad and output the string “Hello world” to LCD display.

Text Books:

1. Andrew Sloss, Dominic Symes, Chris Wright, “ARM System Developer's Guide – Designing and Optimizing System Software”, Elsevier Publication 2012.
2. Lyla B Das, “Architecture, Programming, and Interfacing of Low-power Processors – ARM7,

   Cortex-M”, Cengage Learning India Pvt. Ltd. 2017.

Reference Books:

1. F. Vahid and T. Givargis, “Embedded System Design: A Unified Hardware/Software Introduction”, Wiley India Pvt. Ltd., 2002.
2. A.N. Sloss, D. Symes and C. Wright, “ARM System Developer’s Guide: Design and Optimizing System Software”, Morgan Kaufman Publishers, 2004.

Self-Learning:

Explore emerging trends and practical applications of ARM-based embedded systems, Compare ARM7 and Cortex-M processors with respect to architecture, memory management, power consumption, and applications. Explore other embedded systems.

Course Outcomes:

1. Understand the basics of embedded processors and operating system for different application scenarios (PO: 1; PSO: 2)
2. Describe the building blocks that contribute to the software aspects of embedded system design. (PO:1, 2, 3; PSO1, 2)
3. Design ARM7 programs that includes interrupt structure and pipeline. (PO:1, 2, 3; PSO1, 2)
4. Develop ARM7 assembly code for a given problem (PO:1, 2, 3; PSO1, 2)
5. Compare Cortex-M processors in terms of architecture, memory model and interrupt structure
(PO:1, 2)

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab Record	10	CO1, CO2, CO3, CO4, CO5
Lab test	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

Probability, Statistical Inference and Optimization	
Course Code: IS42	**Credits: 2:1:0:2**
Pre – requisites: Basic Probability	**Contact Hours: 30L+15T+30S**
Course Coordinator: Dr. Govindaraju M V	

Course Contents

Unit I

Statistics: Curve fitting by the method of least squares, fitting linear, quadratic and geometric curves, correlation and regression. Coefficient of determination, or R-squared value. Multivariable curve fitting, multiple correlation and regression.

Discrete Probability Distributions: Random variables, Binomial distribution, Poisson distribution.

- **Pedagogy/Course delivery tools:** :Chalk and talk, Power Point Presentation, Videos
- **Links** :https://nptel.ac.in/courses/111/105/111105035/

Unit II

Continuous Probability Distributions Uniform distribution, Exponential distribution, Gamma distribution and Normal distribution.

Joint probability distribution: Joint probability distribution (both discrete and continuous), conditional probability, conditional expectation.

- **Pedagogy/Course delivery tools:** Chalk and talk, PowerPoint Presentation, Videos
- **Links:**
  - https://nptel.ac.in/courses/111/105/111105035/
  - https://nptel.ac.in/courses/111/107/111107119/
  - https://a.impartus.com/ilc/#/course/96151/1112

Unit III

Stochastic Processes: Introduction, Classification of stochastic processes, Discrete time processes, Stationary stochastic processes, Autocorrelation, Ergodicity.

Markov Chain: Introduction to stochastic process, probability vectors, stochastic matrices, regular stochastic matrices, Markov chains, higher transition probabilities, stationary distribution of regular Markov chains and absorbing states, Markov and Poisson processes.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links:**
  - https://nptel.ac.in/courses/111103022
  - https://a.impartus.com/ilc/#/course/96127/452

Unit IV

Sampling and Statistical Inference: Sampling distributions, central limit theorem, concepts of standard error and confidence interval, level of significance, type I and type II errors, one tailed and two tailed tests, Z-test: for single mean, for single proportion and for difference between means, Student's t –test: for single mean and for difference between two means, F – test for equality of two variances, Chi-square test: for goodness of fit and for independence of attributes.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links:** https://nptel.ac.in/courses/111/105/111105035/
  - https://nptel.ac.in/courses/111/107/111107119/

Unit V

Linear Programming: Introduction to linear programming problem (LPP), formulation of the problem, graphical method, general, canonical and standard forms of LPP, simplex method, big-M method, two-phase simplex method and duality in linear programming.

- **Pedagogy / Course delivery tools:** Chalk and talk.
- **Links:** https://nptel.ac.in/courses/111104027

Textbooks

1. Kreyszig, E. (2025). Advanced Engineering Mathematics (11th ed.). Wiley-India Publishers.
2. Walpole, R. E., Myers, R. H., Myers, S. L., & Ye, K. (2022). Probability and Statistics for Engineers and Scientists (9th ed.). Pearson Education, Delhi.
3. Veerarajan, T. (2017). Probability, Statistics and Random Processes (3rd ed.). Tata McGraw-Hill Education.
4. Swarup, K., Gupta, P. K., & Mohan, M. (2014). Operations Research. Sultan Chand & Sons Publishers.

Reference Books:

1. Grewal, B. S. (2024). Higher Engineering Mathematics (45th ed.). Khanna Publishers.
2. Ross, S. M. (2009). Probability models for Computer Science (13th ed.). Academic Press, Elsevier.
3. Spiegel, M. R., Schiller, J., & Srinivasan, R. A. (2013). Probability and Statistics (4th ed.). Schaum's Outlines. McGraw-Hill.
4. Trivedi, K. S. (2008). Probability & Statistics with Reliability, Queuing and Computer Science Applications (2nd ed.). John Wiley & Sons.
5. Johnson, R. A., & Miller, I. (2015). Miller & Freund's Probability and Statistics for Engineers (8th ed.). Pearson Education India.

Self Study:

Understand the real world applications of statistics and probability in sampling theory, stochastic and Markovian processes and linear programming in optimization

Course Outcomes

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

1. Apply the method of least squares for single and multivariable curve fitting, analyze correlation/regression, and evaluate data using discrete probability distributions. (PO:1, 2; PSO: 1, 3)
2. Analyze continuous probability distributions and evaluate joint, conditional, and expected values for multi-variable random scenarios. (PO:1, 2; PSO: 1, 3)
3. Examine the properties of stochastic processes and determine steady-state behaviors using Markov chains and transition probability matrices. (PO:1, 2; PSO: 1, 3)
4. Choose and apply appropriate statistical significance tests (Z, t, F,and χ^2) to make valid inferences about a population from sample data. (PO:1, 2; PSO: 1, 3)
5. Formulate and solve Linear Programming Problems using graphical, simplex, dual, and artificial variable techniques for optimal resource allocation. (PO:1, 2; PSO: 1, 3)

Tutorial Classes

Session No.	Topic	No. of hours
1	Problems on fitting linear, quadratic, and geometric (exponential) curves using the method of least squares; calculating and interpreting the coefficient of determination (R^2).	2 hours
2	Problems involving multivariable curve fitting, calculating multiple correlation coefficients, and establishing multiple linear regression equations.	2 hours
3	Practical applications and probability calculations using Binomial and Poisson distributions for discrete random variables.	2 hours
4	Evaluating probabilities, means, and variances using Uniform, Exponential, Gamma, and Normal distributions (including standard Z-score table usage).	2 hours
5	Solving marginal and joint probability mass/density functions for both discrete and continuous bivariate random variables.	2 hours
6	Problems on calculating conditional distributions, conditional covariance, and conditional mathematical expectations.	2 hours
7	Problems on classifying stochastic processes, testing for stationarity, calculating autocorrelation functions, and evaluating ergodicity.	2 hours
8	Constructing probability vectors and stochastic matrices; calculating higher-order transition probabilities	2 hours
9	Finding the stationary distribution of regular Markov chains and solving problems related to absorbing states, Markov, and Poisson processes.	2 hours
10	Numerical problems applying the Central Limit Theorem (CLT), calculating standard error, and constructing confidence intervals for means and proportions	2 hours
11	Hypothesis testing (one-tailed and two-tailed) for single means, proportions, and differences between means using Z-tests and Student's t-tests. Identifying Type I and Type II errors	2 hours
12	Running F-tests for equality of two variances, and Chi-square (χ^2) tests for goodness of fit and independence of attributes	2 hours
13	Formulating real-world optimization problems into LPP models; converting equations between canonical and standard forms; solving 2-variable problems using the graphical method	2 hours
14	Setting up simplex tableaus and solving standard optimization linear programming problems using the iterative Simplex algorithm.	2 hours

15	Solving complex LPPs using the Big-M and Two-Phase simplex methods; constructing and solving the Dual of a given primal LPP	2 hours

Course Assessment and Evaluation:

Continuous Internal Evaluation: 50		
Assessment Tool	**Marks**	**Course Outcomes**
Internal test – I	30	CO1, CO2, CO3
Internal test – II	30	CO3, CO4, CO5
Average of the two internal tests will be taken for 30 marks.		
Other components	**Marks**	**Course Outcomes**
Quiz	10	CO1, CO2, CO3
Assignment	10	CO3, CO4, CO5
Semester End Examination (SEE)		
Course end examination(Answer any one question from each unit – Internal choice)	100	CO1, CO2, CO3, CO4, CO5

Design and Analysis of Algorithms	
Course Code: IS43	**Credits: 3:0:0:3**
Pre – requisites: NIL	**Contact Hours: 45L+45S**
Course Coordinator: Krishna Raj	

Course Contents

Unit 1

Foundations of Algorithms: Role of Algorithms in Computing, Growth of Functions, Asymptotic Notation, Analyzing Algorithms, Standard Notations and Common Functions. Divide-and-Conquer: Recurrences, Substitution Method, Recursion-Tree Method, Master Method. Probabilistic Analysis and Randomized Algorithms: Indicator Random Variables, Randomized QuickSort, Randomized Algorithms for Matrix Operations.

- **Pedagogy**: Conceptual Learning using NPTEL video lectures and problem-solving sessions
- **Links**: https://nptel.ac.in/courses/106101060

Unit 2

Sorting and Order Statistics: Heaps and HeapSort, Priority Queues, QuickSort, Sorting in Linear Time - Counting Sort, Radix Sort, Bucket Sort. Medians and Order Statistics: Selection Algorithms and their Analysis.

- **Pedagogy**: Analytical Learning through NPTEL videos, illustrative case problems, and algorithm tracing
- **Links**: https://nptel.ac.in/courses/106101060

Unit 3

Amortized Analysis and Advanced Algorithmic Techniques: Aggregate Analysis, Accounting Method, Potential Method. Union-Find / Disjoint Set Union with Path Compression and Union by Rank. Fibonacci Heaps and their applications in graph algorithms. Applications of Amortized Analysis to data structure operations and network optimization problems.

- **Pedagogy**: Visualization-based Learning with animated NPTEL videos and hands-on paging examples
- **Links**: https://nptel.ac.in/courses/106101060

Unit 4

Dynamic Programming: Matrix Chain Multiplication, Longest Common Subsequence, Optimal Binary Search Trees.

Greedy Algorithms: Activity Selection, Huffman Codes, Minimum Spanning Trees using Kruskal and Prim, Dijkstra's Algorithm.

- **Pedagogy**: Structure-oriented Learning using NPTEL lectures and whiteboard implementation exercises
- **Links**: https://nptel.ac.in/courses/106101060

Unit 5

Graph Traversals: BFS, DFS, Topological Sort, Strongly Connected Components. Shortest Paths: Bellman-Ford, Dijkstra revisited, Floyd-Warshall, Johnson's Algorithm. Maximum Flow: Ford–Fulkerson, Edmonds-Karp.

Computational Complexity: Introduction to P, NP, NP-Completeness, Approximation Algorithms.

- **Pedagogy**: Real-world Case-based Learning with NPTEL content and practical scenario discussions
- **Links**: https://nptel.ac.in/courses/106101060

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms,**"** Fourth Edition, MIT Press, 2022.

Reference Books:

1. Jon Kleinberg, Éva Tardos, "Algorithm Design," 2nd Edition, Pearson, 2022.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.

Self-Learning:

Case Study: Analyze the use of a greedy algorithm in a real-world application like GPS navigation. Examine a scenario where an inefficient algorithm caused problems and propose a better solution.

Project Based Learning: Create a visualizer tool to demonstrate how sorting algorithms work. Build a simple recommendation engine using graph algorithms like BFS or DFS.

Simulation: Simulate Kruskal's Algorithm to find a minimum spanning tree in a network. Use a simulation to visualize the execution of a backtracking algorithm like the N-Queens Problem

Course Outcomes:

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

1. Analyze algorithms using asymptotic notation, recurrences, and probabilistic tools. (PO:1, 2, 4; PSO:1)
2. Apply sorting, selection, and searching techniques efficiently. (PO:1, 3, 5; PSO:1)
3. Employ advanced data structures and amortized analysis to optimize algorithms. (PO:1, 3, 5; PSO:1)
4. Implement algorithmic design paradigms such as dynamic programming and greedy methods. (PO:1, 3, 5; PSO:1)
5. Apply graph algorithms, flow techniques, and complexity concepts to real-world computational problems. (PO: 2, 4, 11; PSO:1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Quiz	10	CO1, CO2, CO3, CO4, CO5
Simulation	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

Operating Systems	
Course Code: IS44	**Credits: 3:0:0:3**
Pre – requisites: NIL	**Contact Hours:45L + 45S**
Course Coordinator: Dr.Jagadeesh Sai D	

Course Contents

Unit 1

Operating Systems Overview: Operating System Structure, Operating System Operations, Process Management, Memory Management, Storage Management, Protection and Security. Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication, Process Scheduling: Basic Concepts, Scheduling Criteria -Scheduling Algorithms

- **Pedagogy**: **Conceptual Learning** using NPTEL video lectures and problem-solving sessions
- **Links**: https://archive.nptel.ac.in/courses/106/105/106105214/

Unit 2

Synchronization: Background, The Critical-Section Problem and its solutions, Synchronization Scheduling Algorithms, Semaphores Classic Problems of Synchronization, Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock detection, Recovery from deadlock.

- **Pedagogy**: Analytical Learning through NPTEL videos, illustrative case problems, and algorithm tracing
- **Links**: https://archive.nptel.ac.in/courses/106/105/106105214/

Unit 3

Memory-Management Strategies: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table. Virtual-Memory Management: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames.

- **Pedagogy**: Visualization-based Learning with animated NPTEL videos and hands-on paging examples
- **Links**: https://archive.nptel.ac.in/courses/106/105/106105214/

Unit 4

File System: File Concept, Access Methods, Directory and Disk Structure, File System Mounting, File Sharing, Implementing File Systems: File, System Structure, File-System Implementation, Allocation Methods, Free space Management.

- **Pedagogy**: Structure-oriented Learning **using** NPTEL lectures and whiteboard implementation exercises
- **Links**: https://archive.nptel.ac.in/courses/106/105/106105214/

Unit 5

Mass Storage Structure: Overview, Disk Structure, Disk Attachment, Disk Scheduling. System Protection: Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control.

- **Pedagogy**: **Real-world Case-based Learning** with NPTEL content and practical protection scenario discussions
- **Links**: https://archive.nptel.ac.in/courses/106/102/106102132/

Text Books:

1.Operating System Concepts, 10th Edition by Abraham Silberschatz, Greg Gagne, Peter B. Galvin. 2018 Edition, Wiley Publications.

Reference Books:

1. Modern Operating Systems, 4th Edition by Andrew S. Tanenbaum, Herbert Bos. 2014 Edition, Pearson.
2.Operating Systems: Internals and Design Principles, 9th Edition by William Stallings. 2018 Edition, Pearson.

Self-Learning: Case Study on Linux

Linux Architecture – Kernel vs Shell, System Boot Process

Process Management in Linux – ps, top, nice, kill, process creation using fork, exec

Synchronization in Linux – Semaphores, inter-process communication (ipcs, ipcrm), race conditions

Memory Management – Use of free, vmstat, /proc/meminfo, paging, virtual memory

File System Structure – Linux file hierarchy, df, du, permissions (chmod, chown, setfacl)

Storage Management – Disk scheduling via iostat, partitioning with fdisk, filesystem mounting

Security & Protection – User/group management, access control via ACLs (getfacl, setfacl)

Course Outcomes:

1. Understand the structure and functionality of the operating system and apply CPU scheduling algorithms for the given problem. (PO:1, 2, 3, 9, 10; 11; PSO:2)
2. Analyze the various algorithms for process synchronization and deadlock handling. (PO:1, 2; PSO2)
3. Apply the various techniques for memory management. (PO:1, 2, 3, 9, 10; 11; PSO:2)
4. Understand file and secondary storage management strategies. (PO:1, 2, 3, 9, 10; 11; PSO:2)
5. Describe the need for information protection mechanisms and the working of modern operating system (PO:1;PSO:2)

Course Outcomes:

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

1. Design the experiment for the given problem using various standard algorithmic techniques. (PO 1,2,3, 9, 10; PSO 1)
2. Develop the solution for the given real world problem. (PO 1, 2, 3, 4, 9, 10; PSO 1)
3. Analyze the results and produce substantial written documentation (PO 1, 2, 4, 9, 10; PSO1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Other Components		
Conduction, Calculation, Record and Viva	30	CO1, CO2, CO3
Semester-End Examination (SEE)	50	CO1, CO2, CO3

Data Analytics Lab	
Course Code: ISL47	**Credits: 0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L**
Course Coordinator: Dr. Lincy Meera Mathews	

Course Contents

Unit 1

Part –A

1. Exploring and Visualizing Sales Data. Import a sample sales dataset into Tableau. Create a dashboard that includes the following visualizations:
   a) A bar chart showing total sales by product category.
   b) A line chart showing monthly sales trends over the past year.
   c) A geographic map showing sales distribution by region.
   d) Use a dual-axis chart or side-by-side bar charts to compare sales and return rates across regions

2. Using a dataset on customer purchases, create a dashboard with the following advanced visualizations:
   a) A scatter plot showing the relationship between customer age and total spending.
   b) A heat map illustrating product sales intensity by store location.
   c) A dual-axis chart comparing sales and profit margins over time.
   d) Create a bar chart or treemap showing total sales revenue by product and region.

3. Analysing Sales Performance: Create various visualizations to analyze the sales performance.
   a) Create a line chart to show sales trends over time.
   b) Create a bar chart to compare sales across different regions.
   c) Use a pie chart to display the distribution of sales among different product categories.
   d) Create a dashboard combining these visualizations to provide an overall view of sales performance.

Part –B

1. Dataset: sklearn.datasets.load_iris()
   a) Explore the dataset and visualize feature distributions using simple plots.
   b) Build a classification model using K-Nearest Neighbors (KNN).
   c) Evaluate the model using Accuracy and Confusion Matrix.

2. Dataset: sklearn.datasets.load_breast_cancer()
   a) Explore the dataset and identify important features using basic statistics.
   b) Build a classification model using Logistic Regression.
   c) Evaluate the model using Accuracy, Precision, Recall, and F1-score

3. Dataset: sklearn.datasets.load_wine()
   a) Explore the dataset and visualize class-wise feature distributions.
   b) Build a classification model using Decision Tree.
   c) Evaluate the model using Accuracy and Confusion Matrix.

4. Dataset: sklearn.datasets.fetch_california_housing()
   a) Explore the dataset and visualize relationships between features and target.
   b) Build a regression model using Linear Regression.
   c) Evaluate the model using R^2 and RMSE.

5. Dataset: sklearn.datasets.load_diabetes()

a) Explore the dataset and visualize feature trends.
b) Build a regression model using Ridge Regression.
c) Evaluate the model using R^2 and MAE.

6. Dataset: sklearn.datasets.load_digits()
   a) Visualize sample digit images from the dataset.
   b) Build a classification model using Support Vector Machine (SVM).
   c) Evaluate the model using Accuracy and Confusion Matrix.

7. Dataset: sklearn.datasets.make_blobs()
   a) Generate a synthetic dataset and visualize data points.
   b) Apply K-Means clustering to group the data.
   c) Evaluate clustering using Silhouette Score.

8. Dataset: sklearn.datasets.make_moons()
   a) Generate a synthetic dataset and visualize the structure of data.
   b) Apply DBSCAN clustering.
   c) Interpret the clusters and noise points using visual plots.

9. Dataset: sklearn.datasets.load_iris()
   a) Use only feature data and ignore labels.
   b) Apply Hierarchical Clustering.
   c) Visualize clusters using dendrogram and scatter plots.

10. Dataset: sklearn.datasets.load_wine() (Dimensionality Reduction)
    a) Explore the dataset features.
    b) Apply Principal Component Analysis (PCA).
    c) Visualize the data in 2D using the first two principal components and interpret results.

Textbooks

1. Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers - How to think like a computer scientist: learning with python 3. Green Tea Press, Wellesley, Massachusetts, 2020 https://media.readthedocs.org/pdf/howtothink/latest/howtothink.pdf

Reference book:

1. Al Sweigart - Automate the Boring Stuff with Python, 2nd edition: Practical Programming for Total Beginners", 2 nd edition, No Starch Press, 2022. (Available under CC-BY-NC-SA license at https://automatetheboringstuff.com/) 2. Kyla McMullen, Elizabeth Matthews and June Jamrich Parsons - Programming with Python, Cengage, 2023.

Self-Learning (Programming Assignment/ Case Study/ Project Based Learning/ MOOC based learning / GATE oriented aptitude test / Simulation): Programming.

Course outcomes:

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

1. Understand and implement various machine learning procedures algorithms using appropriate datasets. (PO: 1, 2, 3, 4, 5; PSO: 1, 3)

2. Design and develop Java/Python programs for supervised, unsupervised, and ensemble learning

methods. (PO: 1, 2, 3, 4, 5; PSO: 1, 3)

3. Apply machine learning techniques to analyze and solve real-world problems. (PO: 1, 2, 3, 4, 5; PSO:1, 3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	Marks	Course outcomes attained
Internal Test	20	CO1, CO2, CO3
Other Components		
Conduction, Calculation, Record and Viva	30	CO1, CO2, CO3
Semester-End Examination (SEE)	50	CO1, CO2, CO3

Design and Analysis of Algorithms Laboratory	
Course Code: ISL48	**Credits: 0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L**
Course Coordinator: Krishna Raj	

Course Contents

General Instruction:
For each experiment, students are required to design and implement the algorithm, demonstrate the working through step-by-step tracing on sample input and evaluate its performance by analyzing the time complexity.

1. An online retail platform must arrange thousands of customer orders by order ID before shipping. Design an efficient sorting approach using arrays, ensuring stable ordering for large datasets.

2. A courier service wants to optimize package sorting for delivery by randomizing the partitioning step to avoid worst-case delays. Use array-based random partitioning to achieve efficient average performance.

3. An airline system maintains passenger boarding priorities based on ticket class and time of check-in. Implement a priority queue using a binary heap to order passengers for boarding efficiently.

4. An exam evaluation system needs to quickly sort thousands of student marks, all within the fixed range of 0 to 100. Apply a linear-time counting technique using arrays to produce the sorted list.

5. A hospital analytics dashboard requires finding the median patient waiting time from thousands of entries recorded daily. Use a selection procedure on an unsorted array to determine the median in guaranteed linear time.

6. A social networking platform must efficiently track friend groups as users form new connections. Implement a disjoint-set data structure with union by rank and path compression to dynamically manage merging groups.

7. A city navigation system computes the shortest routes between intersections for thousands of vehicles. Implement a graph representation with adjacency lists and use a heap-based priority queue optimized by Fibonacci heaps to improve shortest path calculations.

8. A video processing engine must determine the most efficient sequence for multiplying a chain of matrices representing video transformations. Use dynamic programming with a tabular approach to minimize the total number of scalar multiplications.

9. A plagiarism detection system compares two student essays to find the longest matching sequence of words. Use dynamic programming with a two-dimensional table to compute the longest common subsequence between the two text strings.

10. A telecom service provider compresses text messages to reduce transmission costs. Construct a binary tree based on character frequencies and apply a greedy encoding strategy to generate efficient prefix codes for the text.

11 A file management system must explore all files and folders in a directory structure to build dependency

maps. Model the directory as a graph and traverse it using depth-first search to record discovery and finishing times.

12. A city water supply network distributes water from a main reservoir to multiple zones through pipelines of limited capacity. Model the network as a directed flow graph and compute the maximum possible water flow from source to sink.

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms,**"** Fourth Edition, MIT Press, 2022.

Reference Books:

1. Jon Kleinberg, Éva Tardos, "Algorithm Design," 2nd Edition, Pearson, 2022.

2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.

Course Outcomes:

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

1. Design the experiment for the given problem using various standard algorithmic techniques. (PO1, PO2, PO3, PO9, PO10, PSO-1)
2. Develop the solution for the given real world problem. (PO1, PO2, PO3, PO4, PO9, PO10, PSO1)
3. Analyze the results and produce substantial written documentation (PO1, PO2, PO4, PO9, PO10, PSO1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test	20	CO1, CO2, CO3
Other Components		
Conduction, Calculation, Record and Viva	30	CO1, CO2, CO3
Semester-End Examination (SEE)	50	CO1, CO2, CO3

Unix System Programming Lab	
Course Code: ISL49	**Credits: 0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L**
Course Coordinator: Dr. Yogish H K	

Course Content

Part – A

1. Write a shell program that creates a student database file sdb.txt, which consists of sid, sname, ssem, dept. Display only Student Name, Department, and frequency of occurrences of students belonging to respective departments.
2. Write a shell program which takes two file names as command line arguments, if their contents are the same then remove the second file.
3. Write a shell program that takes a filename as a command line argument and reports whether it is a directory, a regular, a character device, a block device, a link or a pipe file.
4. Write a shell program that accepts one or more file names as arguments and converts all of them to uppercase, provided they exist in the current directory.
5. Write a shell script that accepts two file names as command line arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permission and otherwise output each file name followed by its permissions.
6. Write a shell program to do the followings on strings, find length of a string, whether string is NULL, and two strings are equal or not.
7. Write a shell program to print the command line arguments in reverse order.
8. Write a shell program to print the given number in reversed order.
9. Write a shell program to print the first 25 Fibonacci numbers.
10. Write a shell program to print the prime numbers between the specified range.
11. Write a shell program to search the given key element using linear search.
12. Write a shell program to find the largest of three numbers using a user defined function.

Recommended Resources / Links:

1. NPTEL playlist - Linux Programming & Scripting by Anand Iyer(IIT Madras)
   https://www.youtube.com/playlist?list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ
2. NPTEL playlist – Basics of Shell Programming IIT Bombay
   https://www.youtube.com/watch?v=h5Labe97yWM
   https://www.youtube.com/watch?v=rRGCGZ6OHw8&list=PLbMVogVj5nJRa3VKt_eyZdJ_DitCz1cvQ&index=2
3. NPTEL playlist – IISc UNIX and Shell Scripting
   https://archive.nptel.ac.in/courses/106/108/106108101/

PART- B

1. Write a C/C++ program to implement UNIX commands ln, mv, rm commands using APIs.
2. Write a program in C/C++ to display the contents of a named file on the standard output device and write to another file.
3. Write a C/C++ program that reads every 100th byte from the named file, where the file name is given as a command-line argument.
4. Write a C/C++ program to display information of a given file which determines type of file and Inode number, where the file name is given as command line argument.
5. Write a C/C++ program to create a process by using fork ( ) and vfork( ) system call.
6. Write a C/C++ program to demonstrate the process is Zombie and to avoid the Zombie process.
7. Write a C/C++ program to create and demonstrate Orphan Process.
8. Write a C/C++ program to demonstrate a parent process that uses wait () system call to catch the child's exit code.
9. Write a C/C++Program to demonstrate race condition.
10. Write a C/C++ program to implement UNIX system ( ) Function, using APIs.
11. Write a C/C++ program to catch, ignore and accept the signal - SIGINT.
12. Write a C/C++ program to create a pipe from the parent to child and send data down the pipe.

Recommended Resources / Links:

https://archive.nptel.ac.in/courses/106/106/106106144/

Text Books:

1. Sumitabha Das., Unix Concepts and Applications., 4th Edition., Tata McGraw Hill , 2017
2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education.

Reference Books:

1. Unix System Programming Using C++ - Terrence Chan, PHI,
2. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson.

Course Outcomes (COs):

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

1. Design Shell Scripts to solve given real word modules. (PO:1,2, 3,9,10; PSO1)
2. Apply an appropriate APIs to handle various file related operations. (PO:1, 2, 3, 4, 5; PSO:1)
3. Apply an appropriate APIs to manage and control the processes. (PO:1, 2, 3, 4, 5; PSO:1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**

Internal Test	20	CO1, CO2, CO3
Other Components		
Conduction, Calculation, Record and Viva	30	CO1, CO2, CO3
Semester-End Examination (SEE)	**50**	**CO1, CO2, CO3**

Artificial Intelligence	
Course Code: ISE451	**Credits: 3:0:0:3**
Pre – requisites: Introduction to AI	**Contact Hours: 45L + 45S**
Course Coordinator: Dr. Anitha P	

Course Contents

Unit 1

Adversarial Search: Optimal Decisions in Games, Heuristic Alpha-Beta Tree Search. **Logical Agents:** Knowledge Based Agents, The Wumpus World, Logic, Propositional Logic: A very simple logic (wumpus world problem)**.**

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**: https://nptel.ac.in/courses/112103280 , https://nptel.ac.in/courses/106106126

Unit 2

First-Order Logic: Representation Revisited**,** Syntax and semantics of First-Order Logic, Using First - Order logic, Knowledge Engineering in First-Order Logic.
Inference in First Order Logic: Propositional Versus First Order Inference, Unification, Forward Chaining, Backward Chaining, Resolution.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**: https://onlinecourses.nptel.ac.in/noc25_cs07/preview

Unit 3

Automated Planning: Definition of Classical Planning, Algorithms for Classical Planning.
Robotics: Robot, Robot Hardware, Application Domains**.**

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**:https://nptel.ac.in/courses/106105077

Unit 4

Introduction to agents and their world: Defining the concept of agents, Differentiating the components of an agent, Analyzing the rise of the agent era: Why agents, Peeling back the AI interface, Navigating the agent landscape.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**: https://onlinecourses.nptel.ac.in/noc25_cs159

Unit 5

Harnessing the power of large language models: Mastering the OpenAI API, Exploring open source LLMs with LM Studio, Prompting LLMs with prompt engineering, Choosing the optimal LLM for your specific needs, Exercises. **Engaging GPT assistants**: Exploring GPT assistants through ChatGPT.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**:https://onlinecourses.nptel.ac.in/noc26_cs88

Text Books:

1. Russell, S. J., & Norvig, P. (2021). "Artificial Intelligence: A Modern Approach", Global Edition 4e.
2. Micheal Lanham, M. (2025). "*AI agents in action"*. Simon and Schuster.

Reference Books:

1. Elaine Rich, Kevin Knight, Shivashankar B Nair: Artificial Intelligence, 3rd Edition, Tata McGraw Hill, 2017.
2. Deepak Khemani "Artificial Intelligence", Tata Mc Graw Hill Education 2013.
3. Poole, D. L., & Mackworth, A. K. (2010). Artificial Intelligence: foundations of computational agents. Cambridge University Press.

Self Study:

1. AI in Real-World Applications: One AI application in healthcare, finance, education, cybersecurity, agriculture, autonomous vehicles, or robotics.

2. Prompt Engineering: Experiment with different prompting techniques such as:

- Zero-shot prompting
- Few-shot prompting
- Chain-of-Thought prompting
- Role-based prompting

Course Outcomes:
At the end of the course the student will be able to:

1. Apply adversarial search methods to make optimal game decisions and construct knowledge-based agents using logical reasoning. (PO: 1, 2,3,5,12; PSO: 1,3)
2. Apply knowledge using the syntax and semantics of First-Order Logic, and inference techniques such as unification, forward chaining, backward chaining, and resolution to solve reasoning problems. (PO: 1, 2,3,5,6,12; PSO:1,3)
3. Apply algorithms for classical planning in AI systems and analyze the philosophical, ethical, and safety implications of artificial intelligence. (PO-:1, 2,3,5,8, 12; PSO:1,3)
4. Analyze the emergence of Agent-based AI systems, and interpret the evolving agent landscape by examining the design of modern AI interfaces. (PO:1,2,3,5,8,12; PSO:2).
5. Design effective prompts, evaluate and select appropriate LLMs for given tasks, and interact with GPT-based assistants to solve practical AI problems. (PO:1,2,3,5,8,12; PSO:2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		

Other Components		
Prompt Engineering Activity / AI tool demo / small case study on AI application.	10	CO1, CO2, CO3, CO4, CO5
Report on one of the above components.	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

Operations Research	
Course Code: ISE452	**Credits: 3:0:0:3**
Pre – requisites: Nil	**Contact Hours: 45L + 45S**
Course Coordinator:	

Course Contents

Unit 1

Introduction to Operations Research (OR) and Linear Programming (LP): OR Models, Solving the OR Model, Phases of an OR Study; Modeling with Linear Programming (LP), Two variable LP Model, Graphical LP solution, Solution of a Maximization/Minimization Model, Computer solution with Excel Solver; LP Model in Equation Form, Transition from Graphical to Algebraic Solution; TORA Tool.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**:
  - https://www.youtube.com/watch?v=a2QgdDk4Xjw&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=1
  - https://www.youtube.com/watch?v=pzbSURDWluA&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=2

Unit 2

The Simplex Method: The Simplex Method, Special cases in the Simplex Method, Degeneracy, Alternative optima, Unbounded solutions, Non-existing solutions; Artificial Variable Techniques: Two Phase Method, Big-M method; TORA Tool.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links:**https://www.youtube.com/watch?v=qxls3cYg8to&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=4

Unit 3

Duality and Game Theory: Introduction: Concept of Duality; Definition of Primal Dual Problems; General Rules for Converting any Primal into its Dual; Introduction to Dual Simplex Method; Computational Procedure of Dual Simplex Method; Illustrative Examples. Introduction of Game Theory, Characteristics of Games Theory; Minimax (Maximin) Criterion and Optimal Strategy; Saddle Point, Optimal Strategies and the value of game; Solution of Games with Saddle Point(s); Rectangular Games without Saddle Point; Arithmetic Method for (2x2) Games, Gambit tool for Game theory.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**://www.youtube.com/watch?v=wrkcoRrsPQE&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=10

Unit 4

Project Management by PERT-CPM: Introduction, Applications of PERT/CPM Techniques, Basic Steps in PERT/CPM Techniques; Network Diagram representation, Rules for Drawing Network Diagram, Labeling Fulkerson's „I-J" Rule; Time Estimates and Critical Path in Network Analysis; Project Evaluation and Review Technique; TORA tool

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links**: https://www.youtube.com/watch?v=H58TPQNr2kM

Unit 5

Transportation and Assignment Problems: Mathematical Formulation of transportation problem (TP); Matrix Form of TP; Feasible Solution, Basic Feasible Solution and Optimum Solution; Tabular Representation; Special Structure of Transportation Table and their Problems; Initial Basic Feasible Solution to TP; Moving Towards Optimality; Degeneracy in TP; Unbalanced TP; Mathematical Formulation of Assignment Problem (AP); Fundamental Theorems; Hungarian Method for AP; TORA tool.

- **Pedagogy/Course delivery tools:** Chalk and talk, Power Point Presentation, Videos
- **Links:**
    - https://www.youtube.com/watch?v=Q31jKiEXxdc&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=13
    - https://www.youtube.com/watch?v=RnZnIIksdwU&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=14
    - https://www.youtube.com/watch?v=BUGIhEecipE&list=PLjc8ejfjpgTf0LaDEHgLB3gCHZYcNtsoX&index=16

Text Books:

1. Operations Research: An Introduction – Hamdy A Taha – 11th Edition, Pearson Education, India, 2024.
2. Operations Research – S.D. Sharma –18th Edition, KNRN Publications, 2014.

Reference Book:
1. Introduction to Operations Research – Frederick S. Hillier and Gerald J. Lieberman – 10th Edition, Tata McGraw Hill,2024

Course Outcomes:
At the end of the course students will be able to:

1. Formulate linear programming model for a given problem. (PO:1, 2, 3; PSO: 2, 3)
2. Solve a linear programming model using appropriate methods. (PO:1, 2, 5, 9, 10, 12; PSO:2, 3)
3. Apply game theory to model, analyze and solve the given problems. (PO:1, 2, 3, 5, 9, 10, 12; PSO: 2,3)
4. Analyze a Project network using PERT and CPM techniques. (PO:1, 2, 3, 5, 9, 10, 12; PSO: 2,3)
5. Apply transportation and assignment techniques to solve the given problems. (PO:1, 2, 3, 5, 9, 10, 12; PSO:2,3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test	20	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Gate Based Aptitude Test	10	CO1, CO2, CO3, CO4, CO5
Programming Assignment	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

Cyber Security and Ethical Hacking	
Course Code: ISE453	**Credits: 3:0:0:3**
Pre – requisites: Computer Networks	**Contact Hours: 45H + 45S**
Course Coordinator: Dr.Geetha V	

Course Contents

Unit 1

Introduction Cyber Security – History of Internet – Impact of Internet – CIA Triad; Reason for Cyber Crime – Need for Cyber Security – History of Cyber Crime; Cybercriminals – Classification of Cybercrimes 130 – A Global Perspective on Cyber Crimes; Cyber Laws – The Indian IT Act – Cybercrime and Punishment..

Pedagogy: Chalk & Talk, Power Point Presentations

Links:

1. https://nptel.ac.in/courses/106106248
2. https://nptel.ac.in/courses/106104467

Unit 2

Attacks And Countermeasures OSWAP; Malicious Attack Threats and Vulnerabilities: Scope of Cyber-Attacks – Security Breach – Types of Malicious Attacks – Malicious Software – Common Attack Vectors – Social engineering Attack – Wireless Network Attack – Web Application Attack – Attack Tools – Countermeasures

Pedagogy: Chalk & Talk, Power Point Presentations

Links:

1. https://onlinecourses.nptel.ac.in/noc23_cs127/preview
2. https://onlinecourses.nptel.ac.in/noc23_cs44/preview

Unit 3

Harvester – Whois – Netcraft – Host – Extracting Information from DNS – Extracting Information from E-mail Servers – Social Engineering Reconnaissance; Scanning – Port Scanning – Network Scanning and Vulnerability Scanning – Scanning Methodology – Ping Sweer Techniques – Nmap Command Switches – SYN – Stealth – XMAS – NULL – IDLE – FIN Scans – Banner Grabbing and OS Finger printing Techniques

Pedagogy: Chalk & Talk, Power Point Presentations

Links:

1. https://nptel.ac.in/courses/106106241
2. https://nptel.ac.in/courses/108101585

Unit 4

Intrusion Detection Host -Based Intrusion Detection – Network -Based Intrusion Detection – Distributed or Hybrid Intrusion Detection – Intrusion Detection Exchange Format – Honeypots – Example System Snort

Pedagogy: Chalk & Talk, Power Point Presentations

Links:

1. https://nptel.ac.in/courses/106105194
2. https://nptel.ac.in/courses/106106146

Unit 5

Intrusion Prevention Firewalls and Intrusion Prevention Systems: Need for Firewalls – Firewall Characteristics and Access Policy – Types of Firewalls – Firewall Basing – Firewall Location and Configurations – Intrusion Prevention Systems – Example Unified Threat Management Products.

Pedagogy: Chalk & Talk, Power Point Presentations

Links:

1. http://acl.digimat.in/nptel/courses/video/128108019/L14.html
2. https://www.netacad.com/cybersecurity

Text Books

1. Anand Shinde, "Introduction to Cyber Security Guide to the World of Cyber Security", Notion Press, 2021 (Unit 1)
2. Nina Godbole, Sunit Belapure, "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", Wiley Publishers, 2011 (Unit 1)
3. David Kim, Michael G. Solomon, "Fundamentals of Information Systems Security", Jones & Bartlett Learning Publishers, 2013 (Unit 2)

References

1. Patrick Engebretson, "The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made easy", Elsevier, 2011 (Unit 3) 3. Kimberly Graves, "CEH Official Certified Ethical hacker Review Guide", Wiley Publishers, 2007 (Unit 3)
2. Kimberly Graves, "CEH Official Certified Ethical Hacker Review Guide", Wiley Publishers, 2007 (Unit 3)
3. William Stallings, Lawrie Brown, "Computer Security Principles and Practice", Third Edition, Pearson Education, 2015 (Units 4 and 5)

Course Outcomes:

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

1. Explain the basics of cyber security, cybercrime and cyber law. (PO:1, 2, 3, 5, 10; PSO: 1,3)
2. Classify various types of attacks and learn the tools to launch the attacks (PO:1, 2, 3, 5, 10; PSO: 1,3)
3. Apply various tools to perform information gathering. (PO:1, 2, 3, 5, 10; PSO: 1,3)
4. Apply intrusion techniques to detect intrusion. (PO:1, 2, 3, 9, 10; PSO: 1,3)
5. Apply intrusion prevention techniques to prevent intrusion. (PO:1, 2, 3, 5, 10; PSO: 1,3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test	20	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Gate Based Aptitude Test	10	CO1, CO2, CO3, CO4, CO5
Programming Assignment	10	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

UI/ UX	
Course Code: ISAEC461	**Credits: 1:0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L + 15S**
Course Coordinator: Charunayana	

Course Contents

Unit 1

The importance of the user interface: Defining the user interface, the importance of good design, and the benefits of good design. Characteristics of graphical and web user interfaces: interaction styles, the graphical user interface, the concept of direct manipulation, graphical system : advantages and disadvantages, characteristics of the graphical user interface, the web interface, the merging of graphical business system, principles of user interface design

- **Pedagogy**: Power Point Presentation
- **Links**: https://www.nngroup.com/articles/usability-101-introduction-to-usability/

Unit 2

The user interface design process : usability, Know your user or client: important human characteristic in design, Understand the principles of good interface and screen design: navigation flow, Visually pleasing composition, application and page size, application screen elements

- **Pedagogy**: Power Point Presentation, Experiential Learning
- **Links**: https://www.usability.gov, https://m3.material.io

Unit 3

Develop system menus and navigation schemes: Structures of Menus, Functions of Menus, Content of Menus, Phrasing the Menu, Formatting of Menus, Selecting Menu Choices, Web Site Navigation, and Kinds of Graphical Menus select the proper kinds of windows: Window Characteristics, components of windows, Window Presentation Styles, Types of Windows, Organizing Window Functions, The Web and the Browser.

- **Pedagogy**: Power Point Presentation, Experiential Learning
- **Links**: https://www.figma.com, https://m3.material.io,

Unit 4

Interactive devices: Input Devices, output devices. Choose the proper screen – based controls: Operable Controls, Text Entry/Read-Only Controls, Selection Controls, Combination Entry/Selection Controls, Custom Controls, Presentation Controls, Selecting the Proper Controls.

- **Pedagogy**: Power Point Presentation, Experiential Learning
- **Links**: https://m3.material.io/components/text-fields/overview, https://m3.material.io/components/menus/overview

Unit 5

Organize and layout windows and pages: Organizing and Laying Out Screens. Test, test and retest: Usability, Prototypes, Kinds of Tests, Developing and Conducting a Test.

- **Pedagogy**: Power Point Presentation, Experiential Learning

- **Links**: https://www.nngroup.com/articles/visual-hierarchy-ux-definition/, https://m3.material.io/foundations/layout/understanding-layout/overview

TUTORIALS

Text Books:

1. The essential guide to user interface design, third edition, Wilbert O Galitz.

Reference Books:

1. Rex Hartson, Pardha S Pyla, The UX Book, Process and Guidelines for Ensuring a Quality User Experience, Morgan Kauffman
2. https://m3.material.io/
3. https://www.refactoringui.com/

Self-learning is encouraged through case study analysis and project-based learning activities.

Course Outcomes:

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

1. Understand the importance of User Experience and User Interface design in application development
   (PO: 2, 3, 5, 9, 10; PSO: 1)
2. Select and use appropriate interactive devices and screen-based controls in the design of a user interface, based on the needs and goals of the user. (PO: 2, 3, 5, 9, 10; PSO: 1)
3. Design system menus and navigation schemes that are intuitive and user-friendly. (PO: 2, 3, 5, 9, 10; PSO : 1)
4. Select appropriate types of windows, interactive devices, and screen-based controls to enhance user interaction and improve overall user experience. (PO: 2, 3, 5, 9, 10; PSO: 1)
5. Applying feedback to iterative testing processes, refine and improve the UI design continuously. (PO: 2, 3, 5, 9, 10; PSO: 1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Internal Test	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Case study/Project	20	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100 (Scale down to 50 Marks)	CO1, CO2, CO3, CO4, CO5

GNU Linux System Administration	
Course Code: ISAEC462	**Credits: 1:0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L+15S**
Course Coordinator: Dr. Krishna Raj	

Course Content

Unit I

Introduction to System Administration & Configuration: Requirements and core responsibilities of a Linux System Administrator. Setting up a multifunction server: installation prerequisites, Linux distribution deployment essentials and establishing secure remote logins via SSH. Fundamental network interface configuration, package management modifications and system clock synchronization.

- **Pedagogy/Course Delivery tools:** Conceptual learning via interactive walkthroughs and step-by-step practical demonstrations of server installation and remote terminal management.
- **Links:** https://nptel.ac.in/courses/117106113

Unit II

User Management & Local Network Infrastructure: Administering accounts: creating and managing users and groups, configuring file permissions and enforcing user disk quotas. Introduction to local network services: basic configuration of a DHCP server. Network file sharing: implementing Samba services and understanding distributed filesystems.

- **Pedagogy/Course Delivery tools:** Activity-based learning through account provisioning exercises and network service emulation tasks
- **Links:** https://nptel.ac.in/courses/117106113

Unit III

Web Server & Database Administration: Introduction to the classic LAMP (Linux, Apache, MySQL, PHP) application stack. Administering the Apache Web Server: installation procedures, managing primary configuration files, parsing logfiles and configuring basic SSL/TLS encryption. Relational database integration: initializing and managing a MySQL database

- **Pedagogy/Course Delivery tools:** Project-based sessions focused on deploying a functional LAMP environment to host static or dynamic web services.
- **Links:** https://nptel.ac.in/courses/117106113

Unit IV

Automation & Data Maintenance: Shell Scripting: introduction to bash environment fundamentals, writing automation scripts and executing control structures (conditionals, loops, variables). System backup strategies: creating compressed archives using tar, executing remote data synchronization via rsync and managing MySQL database backups.

- **Pedagogy/Course Delivery tools:** Practical syntax tracing and scripting exercises focused on automating routine administrative tasks
- **Links:** https://nptel.ac.in/courses/117106113

Unit V

Network Infrastructure (DNS) & Virtualization Basics: Domain Name System (DNS) fundamentals: understanding DNS basics, setting up a primary DNS server and editing configuration files. Utilizing BIND tools and basic troubleshooting of BIND server deployments. Introduction to virtualization in the modern enterprise: hypervisor types, the popularity of virtualization and an overview of deploying guest operating systems using Xen or VMware on a Linux host.

- **Pedagogy/Course Delivery tools:** Case-study based learning or diagnostic tracing to analyze DNS lookup resolution paths and virtual machine network isolation

Text Book:

1. Tom Adelstein, Bill Lubanovic, "Linux System Administration," 1st Edition, O'Reilly Media, 2007.

Reference Books:

1. Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Dan Mackin, "UNIX and Linux System Administration Handbook," 5th Edition, Pearson Education, 2017.
2. Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, "Linux in a Nutshell," 6th Edition, O'Reilly Media, 2009.

Self-Learning:

Case Study: Analyze a real-world system outage caused by misconfigured file permissions or lack of user disk quotas. Examine a security breach scenario resulting from weak SSH configurations and propose a secure remediation plan.

Project Based Learning: Build an automated server backup tool using bash shell scripting that archives data with tar and synchronizes it via rsync. Set up a local LAMP environment to deploy and secure a basic dynamic web application.

Simulation: Simulate high-volume network traffic on an Apache web server to analyze log files and resource exhaustion. Use a virtualized sandbox environment to simulate DNS zone transfer failures and trace resolution errors**.**

Course Outcomes(COs):

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

1. Configure essential network interfaces, modify distribution package defaults and establish secure remote system access via SSH. (PO 1, 5; PSO 1)
2. Administer user profiles, enforce system quotas and file permissions and provision local network resource sharing using Samba and DHCP (PO 1, 2, 5; PSO 1)
3. Deploy, customize and implement standard security encryption rules on a functional Apache web server integrated with a MySQL database. (PO 1, 3, 5; PSO 1)
4. Design robust automated workflow processes using bash shell scripts and deploy multi-tiered data backup strategies with rsync and tar. (PO 1, 2, 3, 5, 11; PSO 1)
5. Establish core domain name configurations using BIND utilities and distinguish structural characteristics of enterprise virtualization platforms. (PO 1, 2, 5, 11; PSO 1)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Tool Demonstration	10	CO1, CO2, CO3, CO4,CO5
Simulation	10	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

NoSQL Databases	
Course Code: ISAEC463	**Credits: 1:0:0:1**
Pre – requisites: NIL	**Contact Hours: 15L + 15S**
Course Coordinator: Dr. Kusuma S	

Course Content

Unit I

Variety of NoSQL Databases: Data Management with Distributed Databases, ACID and BASE, Four Types of NoSQL Databases

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and PPT.
- **Links:**
  1. https:// youtu.be/F4CzPATviaE?si=p9nZ79yvLICzBG_Y
  2. https://www.coursera.org/learn/introduction-to-nosql-databases

Unit II

Key Value Databases: From Arrays to Key Value Databases, Keys: More than Meaningless Identifiers, Key Value Database Terminology: Key Value Database Data Modelling Terms, Key Value Architecture terms, Designing for Key value, Databases: Key Design and Partitioning.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and PPT.
- **Links:**
  1. https://www.coursera.org/learn/introduction-to-nosql-databases
  2. https://www.datastax.com/guides/what-is-a-key-value-database
  3. https://aws.amazon.com/nosql/key-value/

Unit III

Document Databases: Introduction to Document Databases: Basic Operations on Document Databases, Document Database terminology: Document and Collection Terms, Types of Partition, Data Modelling and Query Processing.

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and PPT.
- **Links:**
  1. https://www.coursera.org/learn/introduction-to-nosql-databases
  2. https://learn.mongodb.com/learning-paths/introduction-to-mongodb

Unit IV

Column Family Databases: Introduction, Differences and Similarities to Key-value and Document Databases, Column Family Database Terminology: Basic Components, Structures and Processes, designing for Column Family Databases: Guidelines for Designing Tables, Guidelines for Indexing

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and PPT.
- **Links:**
  1. https://learn.microsoft.com/en-us/azure/cosmos-db/column-family-introduction
  2. https://www.datastax.com/learn/cassandra-fundamentals

Unit V

Graph Databases: Introduction to Graph Databases, Graphs and network Modelling, Advantages of Graph databases, Graph database Terminology: Elements of graphs, operations on Graphs, Properties, types of Graphs, designing for Graph databases: Querying a graph

- **Pedagogy/Course Delivery tools:** Chalk-and-talk and PPT.
- **Links:**
  1. https://memgraph.com/docs/fundamentals/graph-databases
  2. https://aws.amazon.com/nosql/graph/
  3. https://neo4j.com/developer/get-started/

Text Book:

1. Dan Sullivan, 2016 Edition, NoSQL for Mere Mortals, Pearson publication, ISBN: 978-0134023212

Reference Book:

1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012

Self-Learning: Project-Based Learning through Case Studies

Course Outcomes(COs):

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

1. Explain the emergence and different types of No SQL Databases. (PO1, PO2, PO7, PO9, PO10, PO11, PO12 & PSO2)
2. Understand the modelling concepts for a Key Value Database. (PO1, PO2, PO7, PO 9, PO10, PO11, PO12 & PSO2)
3. Demonstrate the detailed architecture of Document Databases (PO1, PO2, PO5, PO7, PO9, PO10, PO11, PO12 & PSO2).
4. Demonstrate the detailed architecture of Column Family Databases (PO1, PO2, PO5, PO7, PO9, PO10, PO11, PO12 & PSO2).
5. Demonstrate the detailed architecture of Graph Databases. (PO1, PO2, PO5, PO7, PO9, PO10, PO11, PO12 & PSO2).

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal Test-I	30	CO1, CO2, CO3
Internal Test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Case Study/Mini Project	10	CO1, CO2, CO3, CO4,CO5
Report	10	CO1, CO2, CO3, CO4,CO5
Semester-End Examination (SEE)	100 (Scale down to 50)	CO1, CO2, CO3, CO4, CO5