M.Tech.

IN

Computer Science and Engineering

CURRICULUM AND SYLLABI

(Applicable from 2023 admission onwards)

Department of Computer Science and Engineering
NATIONAL INSTITUTE OF TECHNOLOGY CALICUT
Kozhikode - 673601, KERALA, INDIA

The Program Educational Objectives (PEOs) of M. Tech in Computer Science and Engineering

PEO1	The graduates shall have an in-depth knowledge in the fundamentals of Computing, with the ability and confidence to specialize in topics of interest in Computing
PEO2	The graduates shall have the ability to solve problems and critically analyze solutions in the area of interest in Computing
PEO3	The graduates shall have the skillset for using knowledge in Computing for the benefit of society with sound ethical practices and a lifelong interest in contributing to knowledge in the field

Programme Outcomes (POs) & Programme Specific Outcomes (PSOs) of M.Tech. in Computer Science and Engineering

PO1	An ability to independently carry out research/investigation and development work to solve practical problems
PO2	An ability to write and present a substantial technical report/document
PO3	Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program
PO4	Ability to critically analyze solutions, proofs and programs
PO5	Ability to reflect on the thought processes used in arriving at solutions, independently review, communicate and assimilate feedback
PO6	Imbibe the practice of ethics in computing and research
PSO1	An ability to independently carry out research/investigation and development work to solve practical problems in the field of computing, and its applications
PSO2	Ability to critically analyze algorithmic solutions, mathematical proofs and computer programs

CURRICULUM

Total credits for completing M. Tech in Computer Science and Engineering is 75.

COURSE CATEGORIES AND CREDIT REQUIREMENTS:

The structure of M.Tech. programme shall have the following Course Categories:

Sl. No.	Course Category	Minimum Credits
1	Programme Core (PC)	16
2	Programme Electives (PE)	22
3	Institute Elective (IE)	2
4	Projects	35

The effort to be put in by the student is indicated in the tables below as follows:
L: Lecture (One lecture session is of 50 minute duration)
T: Tutorial (One tutorial session is of 50 minute duration)
P: Practical (One practical session is of one hour duration)
O: Outside the class effort / self-study (One unit is of one hour duration)

PROGRAMME STRUCTURE

Semester I

Sl. No.	Course Code	Course Title	L	T	P	O	Credits	Category
1	CS6101E	Mathematical Foundations of Computer Science	3	0	2	7	4	PC
2	CS6102E	Algorithms and Complexity	3	0	2	7	4	PC
3		Institute Elective	2	0/1	0	4/3	2	IE
4	CS6103E	Software Systems lab	1	0	6	5	4	PC
5		Programme Elective 1	3	0	2	7	4	PE
6		Programme Elective 2	3	0	0	6	3	PE
		Total					**21**	**--**

Semester II

Sl. No.	Course Code	Course Title	L	T	P	O	Credits	Category
1	CS6302E	Theoretical Foundations of Machine Learning	3	1	0	8	4	PC
2	CS6196E	Project Phase-1	0	0	0	6	2	PC
3		Programme Elective 3	3	0	0	6	3	PE
4		Programme Elective 4	3	0	0	6	3	PE
5		Programme Elective 5	3	0	0	6	3	PE
6		Programme Elective 6	3	0	0	6	3	PE
7		Programme Elective 7	3	0	0	6	3	PE
		Total					**21**	**--**

Semester III

Sl. No.	Course Code	Course Title	L	T	P	O	Credits	Category
1	CS7197E	Project Phase-2	0	0	0	9	3	PC
2	CS7198E	Project Phase-3	0	0	0	45	15	PC
		Total					**18**	**--**

Semester IV

Sl. No.	Course Code	Course Title	L	T	P	O	Credits	Category
1	CS7199E	Project Phase-4	0	0	0	45	15	PC
		Total					**15**	**--**

List of Programme Electives (2-7)*

(Common for all the M.Tech Programmes of the Department of CSE: CS61, CS62 and CS63)

Sl. No.	Course Code	Course Title	L	T	P	O	Credits
1	CS6104E	Advanced Operating System Design	3	0	0	6	3
2	CS6105E	Algorithms for Big Data	3	0	0	6	3
3	CS6106E	Bioinformatics	3	0	0	6	3
4	CS6108E	Computer Networking	3	0	0	6	3
5	CS6109E	Topics in Image Processing	3	0	0	6	3
6	CS6110E	Pattern Recognition	3	0	0	6	3
7	CS6211E	Topics in Computational Geometry	3	0	0	6	3
8	CS6112E	Topics in Computer Architecture	3	0	0	6	3
9	CS6113E	Topics in Database Design	3	0	0	6	3
10	CS6114E	Topics in Network Systems	3	0	0	6	3
11	CS6115E	Topics in Parameterized Algorithms	3	0	0	6	3
12	CS6116E	Topics in Programming Languages	3	0	0	6	3
13	CS6117E	Topics in Quantum Computing	3	0	0	6	3

14	CS6203E	Topics in Cryptography	3	0	0	6	3
15	CS6204E	Topics in Data Privacy	3	0	0	6	3
16	CS6205E	Topics in Information Security	3	0	0	6	3
17	CS6206E	Systems Security					
18	CS6305E	Advanced Data Structures and Algorithms	3	0	0	6	3
19	CS6306E	Advanced Deep Learning and Computer Vision	3	0	0	6	3
20	CS6307E	AI in Healthcare	3	0	0	6	3
21	CS6308E	Computational Linear Algebra	3	0	0	6	3
22	CS6309E	Computational Optimization Methods	3	0	0	6	3
23	CS6311E	High Performance Computing for AI	3	0	0	6	3
24	CS6312E	Intelligent Agents	3	0	0	6	3
25	CS6313E	Internet of Things	3	0	0	6	3
26	CS6314E	Music Information Retrieval	3	0	0	6	3
27	CS6315E	Neural Networks and Deep Learning	3	0	0	6	3
28	CS6316E	Speech Information Processing	3	0	0	6	3
29	CS6317E	Statistical Foundations of Data Science	3	0	0	6	3
30	CS6318E	Topics in Approximation Algorithms	3	0	0	6	3
31	CS6319E	Topics in Data Mining	3	0	0	6	3
32	CS6320E	Topics in Natural Language Processing	3	0	0	6	3

*Students may also choose any core/elective course of appropriate level offered in the Institute as Programme Electives (2-7), with approval from the Programme Coordinator.

List of Institute Electives

(Common for all the M.Tech Programmes of the Department of CSE: CS61, CS62 and CS63)

Sl. No.	Course Code	Course Title	L	T	P	O	Credits
1	IE6001E	Entrepreneurship Development	2	0	0	4	2
2	ZZ6001E	Research Methodology	2	0	0	4	2
3	MS6147E	Technical Communication and Writing	2	1	0	3	2

List of Program Elective 1 under 'Systems Soft-Core' Basket *

(Common for all the M.Tech Programmes of the Department of CSE: CS61, CS62 and CS63)

Sl. No.	Course Code	Course Title	L	T	P	O	Credits
1	CS6107E	Topics in Compiler Design	3	0	2	7	4
2	CS6122E	Computer Architecture and Design	3	0	2	7	4
3	CS6141E	Distributed Computing and Big Data	3	0	2	7	4

* *Program Elective 1 will not be offered to B. Tech students.*

Notes:

- A student has to earn 75 credits to be eligible for the M. Tech Degree.
- Students aiming for industry internships shall complete their theory/laboratory course requirements in the first two semesters itself.
- A student is eligible for PG Diploma of the Institute, subject to satisfactory completion of the requirements stipulated, as per the ordinances and regulations applicable to their year of admission.

- For “Program Elective 1” in Semester I, students should credit one of the program electives coming under the ‘Systems Soft-Core’ basket shown in the above table in the current curriculum. The department may choose to tag newly proposed electives as ‘Systems Soft-Core’ based on their content.

- As a programme elective (2-7), a student may choose, with the approval from the programme coordinator, any graduate level course offered in the institute, that is not listed as a core course in the M.Tech programme to which he/she is enrolled. In such cases, since the credits for elective courses across various programmes in the institute may vary, the total credits acquired in a semester can also vary. However a student is required to complete all the core courses and also the minimum number of elective courses stipulated in the curriculum under each category, and also acquire a total of 75 credits, for the award of the M.Tech degree.

SYLLABUS

CS6101E MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Pre-requisites: NIL

L	T	P	O	C
3	0	2	7	4

Total sessions: 39L + 26P

Course Outcomes:

CO1: Analyze and solve practical computing problems involving foundations of basic number theoretic and algebraic concepts and applying programming libraries for problem solving.

CO2: Analyze and solve computing problems using fundamental principles of linear spaces and analyze algorithms that use these concepts, and apply programming libraries for problem solving.

CO3: Derive algorithmic solutions for algebraic computing problems.

Divisibility, gcd, prime numbers, fundamental theorem of arithmetic, Congruences, Fermat's theorem, Euler function, primality testing, solution of congruences, Chinese remainder theorem, Wilson's theorem, programming exercises with number theory libraries.

Groups and subgroups, homomorphism theorems, cosets and normal subgroups, Lagrange's theorem, rings, finite fields, polynomial arithmetic, quadratic residues, reciprocity, discrete logarithms, programming exercises with number theory libraries and cryptographic applications.

Vector spaces, basis, dimension, linear maps, rank nullity theorem, duality theorem, Eigenvalues and Eigenvectors, solution to systems of equations, solving of linear systems and Eigenvalue computation with mathematics libraries.

Inner product spaces, orthogonality, orthogonal projections, Hermitian and unitary operators, spectral theorem for Hermitian and unitary operators, singular value decomposition (SVD), Cholesky decomposition, use of mathematical libraries for SVD and Cholesky decomposition, simple experiments involving applications of spectral methods in dimensionality reduction in large data sets and image compression.

References:

1. K. Ireland and R. A. Rosen, *A Classical Introduction to Modern Number Theory*, 2nd ed. Springer, 1998.
2. Niven, H.S. Zuckerman, and Montgomery, *An Introduction to the Theory of Numbers*, 3rd ed. John Wiley and Sons, New York, 1992.
3. S. Axler, *Linear Algebra Done Right*, 2nd ed. Springer, 1997.
4. V. Shoup, *A Computational Introduction to Number Theory and Algebra,* 2nd ed, US: Cambridge University Press, 2008.
5. N. Nassif, J. Erhel and B. Philippe, *Introduction to Computational Linear Algebra,* 1st ed. India: Chapman Hall / C. R. C. Press, 2015.

.

CS6102E ALGORITHMS AND COMPLEXITY

Pre-requisites: NIL

L	T	P	O	C
3	0	2	7	4

Total sessions: 39L + 26P

Course Outcomes:

CO1: Analyze the worst case and average case time/space complexity of an algorithm using techniques such as recurrence analysis, amortized analysis and probabilistic analysis.

CO2: Design algorithms for a given problem using design methodologies such as divide and conquer, dynamic programming, greedy etc., and prove their correctness.

CO3: Classify problems based on their hardness, by applying the notion of reductions.

Analysis: RAM model – Notations, Recurrence analysis - Master's theorem and its proof - Amortized analysis - Advanced Data Structures: B-Trees, Binomial Heaps, Fibonacci Heaps, Disjoint Sets, Union by Rank and Path Compression.

Graph Algorithms and Complexity: Matroid Theory, All-Pairs Shortest Paths, Maximum Flow and Bipartite Matching.

Randomized Algorithms: Fingerprinting, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-Randomization.

Complexity classes - NP-Hard and NP-complete Problems - Cook's theorem NP completeness reductions. Approximation algorithms – Polynomial Time and Fully Polynomial time Approximation Schemes. Probabilistic Complexity Classes, Probabilistic Proof Theory and Certificates.

References:

1. D. C. Kozen, *The Design and Analysis of Algorithms*, 1st ed., US: Springer, 1992.
2. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, 3rd ed., India: PHI, 2010.
3. R. Motwani and P. Raghavan, *Randomized Algorithms*, 1st ed., US: Cambridge University Press, 1995.
4. C. H. Papadimitriou, *Computational Complexity*, 1st ed., US: Addison Wesley, 1994.

CS6103E SOFTWARE SYSTEM LABORATORY

Pre-requisites: NIL

L	T	P	O	C
1	0	6	5	4

Total sessions: 13L + 78P

Course Outcomes:

CO1: Apply scripting tools and programming languages for software development.

CO2: Use documentation tools for preparing documents, articles and presentations.

CO3: Design and build web based solutions using software engineering concepts.

Lecture Sessions:

General purpose programming tools (e.g. Java, C++, use of GUI tools), Web programming tools (e.g. HTML, Java with applets/servlets/JSP/J2EE, CGI, Perl), Development Frameworks (Ruby on Rails, Django).

Tools for good software development process. Make/gmake, source code control systems (e.g. git), debuggers (e.g. gdb) and memory allocation debuggers, Introduction to Integrated Development Environments (e.g. eclipse).

Scripting languages (e,g. Python, Perl). Tools for text processing (e.g. AWK, Python, Lex, Yacc).

Exposure to document creation tools (e.g. Latex), plotting tools (e.g. gnuplot, Excel, pstricks).

The names in parenthesis serve as examples and not strict requirements as the contents may be adapted to software practices and trends at the time of offering the course.

Practical Sessions:

Programming and Data Structures - Review (12 sessions)
Web Development - Using web programming tools. Using framework. (24 sessions)
Software development tools in the Linux Environment (12 sessions)
Scripting languages - Text Processing (18 sessions)
Documentation and Plotting (12 sessions)

References

1. L. Wall, T. Christiansen and R. L. Schwartz, *Programming Perl*, 3rd ed. O'Reilly Media,2000.
2. S. Guelich, S. Gundavaram and G. Birznieks, *CGI Programming with Perl*, O'Reilly Media, 3rd ed. June 2000.
3. M. Summerfield, *Programming in Python 3*, 2rd, Addison Wesley Professional, November 2009.
4. M. Summerfield, *Rapid GUI Programming with Python and Qt*, 1st ed. Prentice Hall, 2009.
5. M. Hart. *Ruby on Rails Tutorial*, Available online at https://www.railstutorial.org/book last accessed 12/4/2018.
6. Wikibook Contributors LaTeX, Wikibooks, available at https://upload.wikimedia.org/wikipedia/commons/2/2d/LaTeX.pdf last accessed 12/4/2018
7. J. Levine, flex & and bison, *O'Reilly Media*, 1st ed., 2009.
8. B. Eckel, *Thinking in Java*, 3rd, Prentice Hall, 2002, Available online at www.bruceeckel.com last accessed 28/3/2010.
9. B. Eckel, *Thinking in C++*, 2nd ed. Vol.1 and Vol.2, Prentice Hall. 2003, Available online at www.bruceeckel.com last accessed 28/3/2010.

CS6302E THEORETICAL FOUNDATIONS OF MACHINE LEARNING

Pre-requisites: NIL

L	T	P	O	C
3	1	0	8	4

Total Sessions: 39L + 13T

Course Outcomes:

CO1: Formulate machine learning problems mathematically, choose appropriate models, and identify suitable learning paradigms.

CO2: Analyze the applicability of stochastic gradient descent algorithm to various convex learning problems.

CO3: Estimate the sample complexity of commonly used hypothesis classes.

PAC Learning

Agnostic probably approximately correct (PAC) learning model, loss functions, true and empirical risk, uniform convergence, no-free-lunch theorem, growth functions and VC dimension, VC dimension of various classes such as hyperplanes and polynomials, the fundamental theorem of PAC learning, non-uniform PAC learning, principles underlying model selection, validation and k-fold cross validation, Tutorial demonstrations on cross validation with sample datasets.

Convex Learning

PAC learnability of hyperplanes, linear programming and perceptron algorithms, convex loss functions, mean squared loss, hinge loss and logistic regression functions, surrogate loss functions, regularized loss functions and stability, learnability of Convex Lipschitz bounded and smooth bounded learning problems using the stochastic gradient descent (SGD) algorithm, sample complexity of convex learning problems. Tutorial demonstrations using software implementing convex learning algorithms on sample datasets.

Applications of SGD

Support vector machines (SVM): dual formulation, SGD implementation of soft-SVM, implementation of soft SVM with kernels, linear multiclass predictors, multiclass SVM with SGD, linear predictors for bipartite ranking,

Feed-forward neural networks, expressive power and sample complexity, SGD and backpropagation. Tutorial demonstrations using software implementing feedforward neural networks and other predictive models with sample data and comparison of performance on sample datasets.

References:

1. S. S. Schwartz and S. Ben David, *Understanding Machine Learning: From Theory To Algorithms,* 3rd ed., India: Cambridge University Press, 2015.
2. M. Mohri, A. Rostamizadeh and A. Talwalkar, *Foundations of Machine Learning*, 3rd ed, India: MIT Press, 2018.
3. C. M Bishop, *Pattern Recognition and Machine Learning,* 1st ed Springer, 2006.

CS6196E PROJECT PHASE-1

Pre-requisites: NIL

L	T	P	O	C
0	0	0	6	2

Course Outcomes:

CO1: Survey the literature on a domain of new research areas and compile findings on a particular topic

CO2: Organize and illustrate technical documentation with scientific rigor and adequate literal standards on the chosen topic strictly abiding by professional ethics while reporting results and stating claims

CO3: Develop and demonstrate aptitude for research and independent learning.

CO4: Demonstrate communication skills in conveying the technical documentation via oral presentations using modern presentation tools.

The objective of this phase of the project is to impart training to the students in collecting materials on a specific area of research interest *(topic)* in the broad domain of Engineering/Science from books, journals and other sources, compressing and organizing them in a logical sequence, and presenting the matter effectively both orally and in written format. The topic chosen by the student shall be approved by the project guide(s) and the evaluation committee. Based on the collected information and acquired knowledge, the student is expected to identify unresolved problems in the domain of the selected topic.

References

1. Relevant literature for the computing problem.

CS7197E PROJECT PHASE-2

Pre-requisites: CS6196E PROJECT PHASE-1

L	T	P	O	C
0	0	0	9	3

Course Outcomes:

CO1: Organize the outcome of the survey of literature as review manuscript on the selected topic of interest

CO2: Organize and illustrate technical documentation with scientific rigor and adequate literal standards on the chosen topic strictly abiding by professional ethics while reporting results and stating claims

CO3: Develop and demonstrate aptitude for research and independent learning.

CO4: Demonstrate communication skills in conveying the technical documentation via oral presentations using modern presentation tools.

The objective of this phase of the project is to impart training to the students in preparing a review manuscript based on the collected materials on the specific topic identified in the broad domain of Engineering/Science from books, journals and other sources, compressing and organizing them in a logical sequence, and presenting the matter effectively both orally and in written format. The manuscript should not be a replica of what is available in the existing literature. The manuscript prepared by the student shall be approved by the project guide(s) and the evaluation committee.

References

1. Relevant literature for the computing problem.

CS7198E PROJECT PHASE-3

Pre-requisites: CS7197E PROJECT PHASE-2

L	T	P	O	C
0	0	0	45	15

Course Outcomes:

CO1: Propose solutions to the computational problem based on a focused literature survey on the topic identified

CO2: Organize and illustrate technical documentation with scientific rigor and adequate literal standards on the chosen topic strictly abiding by professional ethics while reporting results and stating claims

CO3: Develop and demonstrate aptitude for research and independent learning.

CO4: Demonstrate communication skills in conveying the technical documentation via oral presentations using modern presentation tools.

The objective of this phase of the project is to impart training to the students in proposing scientific solutions on a specific topic problem identified in the broad domain of Engineering/Science from books, journals and other sources, compressing and organizing them in a logical sequence, and presenting the matter effectively both orally and in written format. The proposed solution should not be a replica of what is contained in the syllabi of various courses of the M. Tech program. The solution proposed by the student shall be approved by the project guide(s) and the evaluation committee.

References

1. Relevant literature for the computing problem.

CS7199E PROJECT PHASE-4

Prerequisites: CS7198E PROJECT PHASE-3

L	T	P	O	C
0	0	0	45	15

Course Outcomes:

CO1: Reflectively analyze proposed solutions to the identified computing problem.

CO2: Design and develop solutions to the problem and analyze results.

CO3: Prepare a thesis report and defend the thesis on the work done.

CO4: Augment the knowledge base in the chosen area of computing, adhering to ethical practices at every stage.

The student is expected to demonstrate the core competency aimed by this course, i.e., the development of enhancements to the knowledge base in the area of interest in computing. The secondary competencies include the management of time bound projects involving research, analysis of problem complexities, design and development of effective solutions and communication of the project's progress, adhering to ethical practices at every stage. This stage of the project evaluates the state of maturity of these competencies. The student is expected to present two reports at intermediate stages, as well as prepare and defend a thesis on his research work.

References

1. Relevant literature for the computing problem.

IE6001E ENTREPRENEURSHIP DEVELOPMENT

Pre-requisites: NIL

L	T	P	O	C
2	0	0	4	2

Total Lecture Sessions: 26

Course Outcomes:

CO1: Describe the various strategies and techniques used in business planning and scaling ventures.

CO2: Apply critical thinking and analytical skills to assess the feasibility and viability of business ideas.

CO3: Evaluate and select appropriate business models, financial strategies, marketing approaches, and operational plans for startup ventures.

CO4: Assess the performance and effectiveness of entrepreneurial strategies and actions through the use of relevant metrics and indicators.

Entrepreneurial Mindset and Opportunity Identification

Introduction to Entrepreneurship Development - Evolution of entrepreneurship, Entrepreneurial mindset, Economic development, Opportunity Recognition and Evaluation - Market gaps - Market potential, Feasibility analysis - Innovation and Creativity in Entrepreneurship - Innovation and entrepreneurship, Creativity techniques, Intellectual property management. .

Business Planning and Execution

Business Model Development and Validation - Effective business models, Value proposition testing, Lean startup methodologies - Financial Management and Funding Strategies - Marketing and Sales Strategies - Market analysis, Marketing strategies, Sales techniques - Operations and Resource Management - Operational planning and management, Supply chain and logistics, Stream wise Case studies.

Growth and Scaling Strategies

Growth Strategies and Expansion - Sustainable growth strategies, Market expansion, Franchising and partnerships - Managing Entrepreneurial Risks and Challenges - Risk identification and mitigation, Crisis management, Ethical considerations - Leadership and Team Development - Stream wise Case studies.

References:

1. Kaplan, J. M., Warren, A. C., & Murthy V. (Indian Adoption) (2022). *Patterns of entrepreneurship management.* John Wiley & Sons.
2. Kuratko, D. F. (2016). *Entrepreneurship: Theory, process, and practic*e. Cengage learning.
3. Barringer, B. R. (2015). *Entrepreneurship: Successfully launching new ventures*. Pearson Education India
4. Rajiv Shah, Zhijie Gao, Harini Mittal, I*nnovation, Entrepreneurship, and the Economy in the US, China, and India*, 2014, Academic Press
5. Sundar,K.(2022). *Entrepreneurship Developmen*t, 2nd Ed , Vijaya Nichkol Imprints, Chennai
6. E. Gordon,Dr. K. Natarajan., (2017).*Entrepreneurship Development*, 6th Ed, Himalya Publishers,Delhi
7. Debasish Biswas, Chanchal Dey,*Entrepreneurship Development in India*, 2021, Taylor & Francis.

ZZ6001E RESEARCH METHODOLOGY

Pre-requisites: NIL

L	T	P	O	C
2	0	0	4	2

Total Lecture sessions: 26

Course Outcomes

CO1: Explain the basic concepts and types of research

CO2: Develop research design and techniques of data analysis

CO3: Develop critical thinking skills and enhanced writing skills

CO4: Apply qualitative and quantitative methods for data analysis and presentation

CO5: Implement healthy research practice, research ethics, and responsible scientific conduct

Exploring Research Inquisitiveness

Philosophy of Scientific Research, Role of Research Guide, Planning the Research Project, Research Process, Research Problem Identification and Formulation, Variables, Framework development, Research Design, Types of Research, Sampling, Measurement, Validity and Reliability, Survey, Designing Experiments, Research Proposal, Research Communication, Research Publication, Structuring a research paper, structuring thesis/ dissertation

Research Plan and Path

Developing a Research Plan: Reviewing the literature- Referencing – Information sources – Information retrieval – Role of libraries in information retrieval – Tools for identifying literatures – Reading and understanding a research article – Critical thinking and logical reasoning; Framing the research hypotheses, Converting research Question into a Model; Data collection- Types of data-Dataset creation- Primary and Secondary data- Scales of measurement- Sources and collection of data- Processing and analysis of data-Understanding Data-statistical analysis, displaying of data-Data visualization-Data interpretation; Research design- Qualitative and Quantitative Research- Designing of experiments- Validation of experiments- Inferential statistics and result interpretation

Scientific Conduct and Ethical Practice

Plagiarism– Ethics of Research- Scientific Misconduct- Forms of Scientific Misconduct. Plagiarism, Unscientific practices in thesis work-Conduct in the workplace and interaction with peers – Intellectual property: IPR and patent registration, copyrights; Current trends – Usage and ethics of AI tools in scientific research.

References:

1. Leedy, P D, "*Practical Research: Planning and Design",* USA: Pearson, Twelfth ed., 2018.
2. Krishnaswamy, K. N., Sivakumar, A. I., and Mathirajan, M., "*Management Research Methodology"*, Pearson Education, 2006.
3. Tony Greenfield and Sue Greener., *Research Methods for Postgraduates*, USA: John Wiley & Sons Ltd., Third ed., 2016.
4. John W. Creswell and J. David Creswell, "*Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*", USA: Sage Publications, Sixth ed., 2022

MS6174E TECHNICAL COMMUNICATION AND WRITING

Pre-requisites: NIL

L	T	P	O	C
2	1	0	3	2

Total Lecture Sessions: 26

Course Outcomes:

CO1: Apply effective communication strategies for different professional and industry needs.

CO2: Collaborate on various writing projects for academic and technical purposes.

CO3: Combine attributes of critical thinking for improving technical documentation.

CO4: Adapt technical writing styles to different platforms.

Technical Communication

Process(es) and Types of Speaking and Writing for Professional Purposes - Technical Writing: Introduction, Definition, Scope and Characteristics - Audience Analysis - Conciseness and Coherences - Critical Thinking - Accuracy and Reliability - Ethical Consideration in Writing - Presentation Skills - Professional Grooming - Poster Presentations

Grammar, Punctuation and Stylistics

Constituent Structure of Sentences - Functional Roles of Elements in a Sentence - Thematic Structures and Interpretations - Clarity - Verb Tense and Mood - Active and Passive Structures - Reporting Verbs and Reported Tense - Formatting of Technical Documents - Incorporating Visuals Elements - Proofreading

Technical Documentation

Types of Technical Documents: Reports, Proposals, Cover Letters - Manuals and Instructions - Online Documentation - Product Documentation - Collaborative Writing: Tools and Software - Version Control Document Management - Self Editing, Peer Review and Feedback Processes

References:

1. Foley, M., & Hall, D. (2018). *Longman advanced learner's grammar, a self*-study *reference & practice book with answers*. Pearson Education Limited.
2. Gerson, S. J., & Gerson, S. M. (2009). *Technical writing: Process and product*. Pearson.
3. Kirkwood, H. M. A., & M., M. C. M. I. (2013). *Hallidays introduction to functional grammar* (4th ed.). Hodder Education.
4. Markel, M. (2012). *Technical Communication* (10th ed.). Palgrave Macmillan.
5. Tuhovsky, I. (2019). *Communication skills training: A practical guide to improving your social intelligence, presentation, Persuasion and public speaking skills.* Rupa Publications India.
6. Williams, R. (2014). *The Non-designer's Design Boo*k. Peachpit Press.

CS6104E ADVANCED OPERATING SYSTEM DESIGN

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Synthesize the design components of a modern operating System

CO2: Analyze and Implement the conceptual architecture of the OS

CO3: Apply the fundamental and extended OS concepts into designing special purpose operating systems

Process management

Contemporary operating-system structure: process API, Direct execution, CPU scheduling, multi-level feedback scheduling, lottery scheduling, multi CPU scheduling

Memory management

Memory API, segmentation, free space management, paging, TLB, advanced page tables, swapping: mechanism and policies, VM systems

Concurrency and Persistence

Thread API, locks and associated data structures, semaphore, concurrency, event-based concurrency,
Files: directories, and RAID, file system - fast file system (FFS), log structured file system, journaling, Network file system, Andrew file system

Special purpose OS

Security in operating system, distributed systems, network operating system, mobile operating system - Android kernel, boot process, file system, Dalvik and Zygote, platform security

.

References:

1. Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau, "*Operating Systems: Three Easy Pieces*", 1st ed USA: Arpaci-Dusseau Books, 2018
2. Charles Crowley, "*Operating Systems: A Design-Oriented Approach*", 1st ed, USA: McGraw Hill Education, 2017
3. Russ Cox, Frans Kaashoek, and Robert Morris, "*xv6: a simple, Unix-like teaching operating system*", 1st ed MIT, 2006.
4. Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, "*Operating System Concepts",* 8th ed, USA: John Wiley and Sons, 2008.
5. Trent Jaeger, "Operating System Security", 1st ed USA: Morgan & Claypool Publishers, 2008
6. G. Meike and Lawrence Schiefer, "*Inside the Android OS: Building, Customizing, Managing and Operating Android System Services*", 1st ed USA: Addison-Wesley, 2021

CS6105E ALGORITHMS FOR BIG DATA

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze and model algorithmic questions on big data

CO2: Apply modern techniques to address such questions, and analyze resulting algorithms

CO3: Design novel solutions to algorithmic problems

Introduction to Sketching and Streaming

Sketching/Streaming: approximate counting, distinct elements count; impossibility results; frequency moments, "Tug of War" sketch; heavy hitters, CountMin and CountSketch algorithms; Dimension reduction: random projections (Johnson–Lindenstrauss lemma); fast dimension reduction; Binary Hashing Techniques- LSH, MLH

Linear Algebra via Sketching

Numerical linear algebra (via sketching): regression via Sketch-And-Solve, subspace embeddings; low-rank approximation; Compressed sensing: Restricted Isometry Property (RIP), $l1$ minimization; iteration hard thresholding;

Streams for Graphs

Streaming for graphs: spanners, triangle counting; dynamic graph algorithms via `0 sampling; Distribution (statistical hypothesis) testing: uniformity testing, closeness;

Property Testing

Property testing / sublinear-time algorithms: – monotonicity testing; – graph property estimation; – geometric problems; Parallel computation: – MapReduce-like parallel models, sorting; – graph algorithms.

References:

1. S Muthukrishnan, *Data Streams: Algorithms and Applications (Foundations and Trends in Theoretical Computer Science)*, 1st ed Now publishers, 2005
2. R. Motwani and P. Raghavan, *Randomized Algorithms,* 1st ed ., Cambridge University Press, 2004
3. A. Bhattacharyya and Y. Yoshida, *Property Testing: Problems and Techniques*, 1st ed Springer, 2022.
4. D. P. Woodruff, *Sketching as a tool for numerical linear algebra (Foundations and Trends in Theoretical Computer Science)*, 1st ed ., Now Publishers,2014.

CS6106E BIOINFORMATICS

.

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Define and analyze the structure of biomolecules such as DNA, RNA and Protein.

CO2: Analyze Biological information using tools and databases.

CO3: Apply various data structures to represent Biological data for efficiently solving Biological problems.

CO4: Critically analyze different Bioinformatics algorithms, and develop novel and efficient methods for Biological data analysis.

Introduction to molecular biology, gene structure and Information content, molecular biology tools, algorithms for sequence alignment, sequence databases and tools.

molecular phylogenetics, phylogenetic trees, algorithms for phylogenetic tree construction, introduction to Perl programming for bioinformatics.

Introduction to protein structure, algorithms for protein structure prediction, gene expression analysis, microarray, pathway analysis.

Pattern matching Algorithms, bio-data analysis, data mining in bioinformatics, algorithms and data structures for efficient analysis of biological Data, emerging trends in bioinformatics.

References:

1. D. E. Krane and M. L. Raymer, *Fundamental Concepts of Bioinformatics,* 1st ed, Pearson Education, 2003.
2. T. K. Attwood and D. J. Parry-Smith, *Introduction to Bioinformatics*,1st ed, Pearson Education, 2003.
3. A. M Lesk, *Introduction to Bioinformatics*,1st ed, Oxford University Press, 2002.
4. J. M. Claverie and C. Notredame, *Bioinformatics – A Beginner's Guide,* 1st ed, Wiley., 2003.
5. N. C Jones and P. A. Pevzner, *An Introduction to Bioinformatics Algorithms,* 1st ed, MIT Press, 2004.
6. Current Literature.

CS6107E TOPICS IN COMPILER DESIGN

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1**:** Formulate new data-flow analysis to capture the information required for an optimization and build an algorithm for the optimization.

CO2**:** Translate intermediate code to machine code.

CO3**:** Evaluate the efficiency and optimality of a given optimization and suggest improvements.

Review of compiler phases, symbol table structure, intermediate representations. control flow analysis: basic blocks and CFG, dominators and loops.

Data flow analysis: reaching definitions, available expressions, and live variable analysis,
Optimizations: redundancy elimination, dead code elimination, loop optimizations, value numbering.

Static single assignment form (SSA): SSA construction, optimizations on SSA form. register allocation, graph coloring algorithm.

Machine code generation: instruction selection - maximal munch and dynamic programming algorithms, code generation for target machine, code generation for run-time management.

References

1. A. V. Aho, M. S. Lam, R. Sethi, and J. D. Ullman, *Compilers: Principles, Techniques, and Tools*, Pearson Education, 2007.
2. S. Muchnick., *Advanced Compiler Implementation*,1st ed, Morgan Kaufmann Publishers, 1997.
3. A. W. Appel and J. Palsberg, *Modern Compiler Implementation in Java*,1st ed, Cambridge University Press, 2002.
4. Y. N. Srikant and P. Shankar, *The Compiler Design Handbook: Optimization and Machine Code Generation*, 1st ed, CRC Press, 2003.

CS6108E COMPUTER NETWORKING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze and compare new protocols in computer networks.

CO2: Evaluate performance of protocol enhancements using modern tools.

CO3: Propose new solutions to recent problems of interest in literature and compare different possible solutions.

Overview of computer networks, TCP/IP protocol, Application layer protocols, Software defined networking, content distribution, Web 2.0, overlay networks, P2P networks.

TCP extensions for high-speed networks, Transaction-oriented applications. New options in TCP, TCP performance issues over wireless networks, SCTP, DCCP.

IPv6: Why IPv6, Basic protocol, Extensions and options, Support for QoS, Security, Neighbour discovery, Auto-configuration, Routing. Changes to other protocols. Application Programming Interface for IPv6, 6bone.
IP Multicasting, Wide area multicasting, Reliable multicast. Routing layer issues, ISPs and peering, BGP, IGP, Traffic Engineering, Routing mechanisms: Queue management, Packet scheduling. MPLS, VPNs

MAC protocols for high-speed LANS, MANs, Wireless LANs and mobile networks, VLAN. Fast access technologies. Gigabit Ethernet. Multimedia networking, Network management.

References

1. W. R. Stevens, *TCP/IP Illustrated: The Protocols,Vol. 1*: 1st ed, Addison Wesley, 1994.
2. G. R. Wright, *TCP/IP Illustrated: The Implementation Vol. 2*: 1st ed, Addison Wesley, 1995.
3. P. Loshin, *IPv6: Theory, Protocol, and Practice*, 2nd ed., Morgan Kaufmann, 2003.

CS6109E TOPICS IN IMAGE PROCESSING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Use different modern software/tools for implementing image processing applications.

CO2: Design and build image processing algorithms/applications that can be used in domains like Medical imaging, Satellite Imaging and Video Surveillance.

CO3: Analyzing the performance and complexity of various techniques being used to design domain specific algorithms for improving the performance.

CO4: Propose new solutions to issues of current interest computer vision such as object detection, segmentation and classification, based on inferences made from the literature review and analysis.

Introduction - Digital image representation - Fundamental steps in image processing - Elements of digital image processing systems - Digital image fundamentals - Elements of visual perception - A simple image model - Sampling and quantization - Basic relationship between pixels - Image geometry - Image transforms - Introduction to Fourier transform – Discrete Fourier transform - Some properties of 2d-fourier transform (DFT)- Other separable image transforms - Hotelling transform

Image enhancement - Point processing - Spatial filtering - Frequency domain - Image restoration - Degradation model - Diagonalization of circulant and block circulant matrices - Inverse filtering - Least mean square filter.
Image Segmentation: Thresholding: Different types of thresholding methods, Point detection, Edge detection: Different types of edge operators, Line detection, Edge linking and boundary detection, Region growing, Region splitting, Region Merging.

Image compression - Image compression models - Elements of information theory - Error-free compression - Lossy compression - Image compression standards.
Image reconstruction from projections - Basics of projection - Parallel beam and fan beam projection - Method of generating projections - Fourier slice theorem - Filtered back projection algorithms - Testing back projection algorithms

Computer Vision - Introduction to machine vision, Image Classification, Object Detection, Semantic Segmentation, Image registration, Introduction to Machine Learning, Machine learning for Computer Vision tasks, Applications in medical imaging, industry, augmented reality, robotics, autonomous vehicles.

References

1. R. C., Gonzalez and Woods R.E, *Digital Image Processing,* 1st ed. Addison Wesley, 1999.
2. Richard Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed. Springer 2022
3. A. K. Jain, *Fundamentals of Digital Image Processing,* 1st ed. Prentice Hall, Englewood Cliffs, 2002.
4. W. K. Pratt, *Digital Image Processing,* 1st ed. John Wiley, 2002.

CS6110E PATTERN RECOGNITION

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Formulate pattern recognition tasks in relation to fundamental mathematical principles of probability theory, linear algebra and optimization.

CO2: Apply linear classifiers like logistic regression, least squares classifier and perceptron, and nonlinear classifiers like Support Vector Machines and Artificial Neural Networks.

CO4: Design and rate new pattern recognition models for problems of current interest, analyse their performances and compare with existing approaches.

Introduction- Introduction to statistical, syntactic and descriptive approaches, Features and feature extraction, Learning.Bayes Decision theory- Introduction, continuous case, 2-category classification,Minimum error rate classification, Classifiers, discriminant functions, and decision surfaces. Error probabilities and integrals, normal density, Discriminant functions for normal density, Bayes Decision theory Discrete case**.**

Parameter estimation and supervised learning- Maximum likelihood estimation, the Bayes classifier,Learning the mean of a normal density, General bayesian learning.Nonparametric technique- Density estimation, Parzen windows, K-nearest Neighbor estimation, Estimation of posterior probabilities, K nearest neighbor rule.

Multiplayer neural networks- Feed forward operation and classification, Backpropagation algorithm,Error surfaces, Back propagation as feature mapping, Practical techniques for improving backpropagation, Convolutional Neural Networks and Deep Learning**.**

Linear Methods: Linear regression, logistic regression, Principal Component Analysis,Fisher's Linear Discriminant Analysis.Non-linear methods - Kernel Methods - Kernel version of PCA, LDA,SVMs Unsupervised learning and clustering- Mixture densities and identifiably, Maximum likelihood estimates, Application to normal mixtures, Unsupervised Bayesian learning, Data description and clustering, Hierarchical clustering, Low dimensional representation of multidimensional map

References:

1. C. M Bishop, *Pattern Recognition and Machine Learning,* 1st ed Springer, 2006.
2. K. S. Fu, *Syntactic Pattern Recognition and Applications,* 1st ed Prentice Hall, 1982.
3. R. O. Duda, P. E. Hart and D.G Stork, *Pattern Classification,* 2ed ed. John Wiley, 2001.

CS6211E TOPICS IN COMPUTATIONAL GEOMETRY

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Design and develop algorithms and data structures for geometric problems.

CO2: Formulate a geometric solution using randomization as a tool.

CO3: Use CGAL to implement advanced geometric problems.

Art Gallery problem and its associated theorems, Triangulation of a polygon and its theory, Area of a polygon. Polygon partitioning, Monotone partitioning, Trapezoidalization, Plane sweep, Partitioning to monotone mountains. Introduction to Computational Geometric Algorithms Library (CGAL) and openGL and coding of simple programs with visualization using QT.

Convex hull in two dimensions, Algorithms for convex hull with their complexity analysis. Implement Convex Hull algorithms and one application using CGAL & visualization using QT.
Voronoi diagram: Algorithm for construction of Voronoi diagram with its complexity analysis. Delaunay triangulation : Preliminaries and properties, Medial axis transform and its properties. Applications of Voronoi Diagram / Delaunay triangulation / Medial axis transform. Implement one application of Voronoi diagram/ Delaunay triangulation using CGAL & visualization using QT.

Binary space partitions: Definition, basic concepts, construction using randomized algorithm, theorems, CGAL implementation of Painter's algorithm
Robot motion planning: Workspace and configuration space, Point robot, Minkowski sums, Translational motion planning, Quadtrees: Uniform and non-uniform meshes, Quadtrees for point sets, Quadtrees to meshes.

Visibility Graphs: Shortest paths for a point robot, Shortest paths for a translating polygonal robot.
Interval Trees, Priority Search Trees, Segment Trees, Partition trees, Multi-level partition trees. Simplex Range Searching

References:

1. M. de Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*, 2nd ed.(revised), Springer-Verlag, 2000.
2. S. L. Devadoss and J. O'Rourke, *Discrete and Computational Geometry*, 1st ed. Princeton University Press, 2011.
3. K. Mulmuley, *Computational Geometry: An Introduction through Randomized Algorithms*, 1st ed. Prentice-Hall, 1994.

CS6112E TOPICS IN COMPUTER ARCHITECTURE

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Apply different modern tools for performance analysis of design enhancements and analyze the system level implications of design enhancements.

CO2: Critically analyze and compare different architecture level design options, and design solutions.

CO3: Propose new solutions to issues of current interest, and do comparative analysis of different possibilities.

DLP and TLP in architecture

Thread level parallelism: multithreaded processors and multicore processors concept, data level parallelism (DLP) - vector architecture, design issues and implementation, graphical processing unit (GPU)- GPU architecture, programming examples using OpenMP and CUDA

Hardware security

Side channel attacks: prime+probe, flush+reload, mitigation techniques, Prefetcher based attacks, PASS-P scheduling - performance and security tradeoffs,

State-of-the-art architectures

Heterogeneous ISA architectures: scheduling mechanisms in heterogeneous ISAs, performance efficient dynamic core, energy efficient dynamic core, advanced cache replacement policies, application specific integrated circuit architectures (ASIC)

References:

1. J. L. Hennessy and D. Patterson, *Computer Architecture: A Quantitative Approach*, 6th ed., Morgan Kaufmann, 2017.
2. ACM SIGARCH *Computer Architecture* News.
3. The WWW *Computer Architecture* page. http://www.cs.wisc.edu/arch/www/ last accessed 23/3/2016.
4. J. P. Shen and M. H. Lipasti, Modern processor design: *Fundamentals of superscalar architectures*, 1st ed., Waveland Pr Inc, 2013.

CS6113E TOPICS IN DATABASE DESIGN

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions:39

Course Outcomes:

CO1: Analyze articles containing topics of current interest in database design.

CO2: Critically analyze Parallel, Distributed and Object oriented databases, and solve advanced problems in Internet databases with the help of Data Mining Algorithms.

CO3: Experiment with partial and temporal databases like MongoDB, Hadoop GIS, and discuss the concepts of mobile and multimedia databases.

CO4: Design, develop a database project and deploy efficient IT solutions using free and open software to help society.

Parallel and Distributed Databases: Architecture of Parallel Databases, Parallel Query Optimization, Distributed DBMS Architectures, Distributed Query Processing, Distributed Concurrency Control, Distributed recovery.

Internet Databases and Data Mining: XML –QL, Ranked Keyword searches on the Web, Data Mining, Clustering, Similarity Search over Sequences.

Object Oriented Database Systems: User Defined ADTs, Objects, Object Identity and Reference types, Database Design for ORDBMS, OODBMS, Comparison of RDBMS with OODBMS and ORDBMS.

Spatial and Deductive Databases: Spatial and Temporal Databases, Temporal Logic, Spatial Indexes, Introduction to Recursive Queries, Introduction to Mobile Databases, Main Memory and Multimedia Databases

References:

1. R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 3rd ed, Addison Wesley.
2. P. O'Neil and E. O'Neil , *Database Principles, Programming, and Performance*, 2nd ed, Harcourt Asia, Morgan Kaufman.
3. A. Silberschatz , H. F. Korth, and S. Sudarshan , *Database System Concepts*, 1st ed, Tata McGraw Hill, 2003.
4. J. D. Ullman, *Principles of Database Systems*,1st ed, Galgotia Publications,1996.
5. C. J. Date, *An Introduction to Database Systems*, 1st ed, Addison Wesley, 2000.
6. R. Ramakrishnan and J. Gehrke, *Database Management Systems*, 3rd ed, McGraw Hill, 2004.

CS6114E TOPICS IN NETWORK SYSTEMS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: To design and implement the network using state of the art in network protocols and architecture.

CO2: To apply novel ideas in networking through various advanced topics.

CO3: To analyze the networks and to engage in advanced research.

Internetworking

Introduction: History and Context-Packet switching, Architectural Principles, Protocol Stacks and Layering, Names, Addresses, IPv4 Addressing, IPv6 addressing, IP forwarding, IP Packets & Routers, Interdomain Routing, Routing: RIP, OSPF, & BGP, Multicast, IPv6, tunneling, NAT, VPN, Virtual circuits, MPLS. Open Flow, Software Defined Networking

Congestion control

Resource Management: End-to-End Congestion Control, Fair Queueing, Router congestion control. Enterprise and Datacenter Networking: Enterprise Networking, Data Center Networking, The Incast Problem.

Wireless Networking

Wireless Networks Overview and Architectures, Routing in Ad-hoc Networks, Making the Best of Broadcast, WiFi and Mobility.

Application networks

Applications, Naming, and Overlays: Topology, Overlay Networks, Distributed Hash Tables, DNS and the Web, HTTP/3, Web 3.0, Policy Switching, Miscellaneous Topics: Tracing and Prototyping, Multicast, Network Energy Issues, eBPF and BPF

References:

1. Larry Peterson and Bruce Davie, *Computer Networks: A Systems Approach*, 4th ed. (2007)
2. James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, 5th ed. (2010)
3. W. Richard Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, 1st ed.
4. W. Richard Stevens, *Unix Network Programming: Networking APIs: Sockets and XTI* (Volume 1), 1st ed
5. W. Richard Stevens, *Advanced Programming in the Unix Environment*, 1st ed, Addison-Wesley, 1993.

CS6115E TOPICS IN PARAMETERIZED ALGORITHMS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Classify problems into tractable and intractable problems.

CO2: Illustrate the techniques for the design of fixed-parameter tractable algorithms.

CO3: Design and analyze fixed-parameter tractable algorithms for some of the classical NP-Hard graph problems.

Review of complexity classes - P, NP, Co-NP, NP-Hard and NP-complete problems - Strategies for coping with hard algorithmic problems; Exact exponential algorithms and the notion of fixed-parameter tractability.

Parameterizations and Parameterized problems- Kernelization - Formal definitions - Some simple kernels, Crown decomposition, Bounded Search trees - Vertex Cover, Feedback Vertex Set

Iterative Compression - Illustration of the technique - Feedback vertex set - Odd Cycle Transversal - Dynamic programming over subsets – Set cover, Tree structured variants of set cover and Steiner Trees. Randomized methods in Parameterized algorithms – Simple randomized algorithm for Feedback Vertex Set, Color coding algorithm for Longest path.

Trees - narrow grids and dynamic programming - Path and Tree decomposition – Dynamic Programming on graphs of bounded treewidth – Treewidth and Monadic second-order logic, Graph searching, Interval and chordal graphs.

References

1. M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk and S. Saurabh, *Parameterized Algorithms, Springer*, 1st ed, June 2015.
2. R. Niedermeier, *Invitation to Fixed-parameter Algorithms*, 1st ed, Oxford University Press, 2006.
3. R. G. Downey and M. R. Fellows, *Fundamentals of Parameterized Complexity*, 1st ed, Springer,2013.

CS6116E TOPICS IN PROGRAMMING LANGUAGES

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Develop formal semantics for programming language constructs.

CO2: Model Programming Language features using Lambda Calculus.

CO3: Design type systems for language safety.

CO4: Design a programming language (formal semantics and type system) with required constructs

Introduction to programming Languages. untyped arithmetic expressions - syntax and semantics, properties of the language of untyped arithmetic expressions.

Untyped Lambda calculus: syntax, operational semantics, evaluation strategies – programming in Lambda calculus, typed arithmetic expressions: typing relation, type safety.

Simply Typed lambda Calculus: typing relation, properties of the language, type safety.
Extensions: basic types, derived forms, let bindings.

Extensions to Lambda Calculus: pairs, tuples, records, sums, variants, references, exceptions, subtyping, recursive types, polymorphism.

References

1. B. C Pierce, *Types and Programming Languages*, 1st ed. MIT Press, 2002.
2. A. B. Tucker, *Handbook of Computer Science Engineering*, 1st ed. CRC Press, 1996.
3. M. L. Scott, *Programming Languages Pragmatics*, 1st ed. Elsevier, 2004.

CS6117E TOPICS IN QUANTUM COMPUTING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1**:** Analyze the quantum computational model and compare with the classical models of computation with respect to computational power and algorithmic efficiency.

CO2**:** Illustrate and summarize well known quantum algorithms for problems like integer factorization and search.

CO3: Analyze quantum error correction and coding schemes

Foundations:Finite Dimensional Hilbert Spaces – Tensor Products and Operators on Hilbert Space –Hermitian and Trace Operators - Basic Quantum Mechanics necessary for the course.

Model of Computation:Quantum Gates and operators and Measurement – Quantum Computational Model – Quantum Complexity – Schemes for Physical realization (Only peripheral treatment expected).

Algorithms and Complexity Shor's Algorithm – Application to Integer Factorization – Grover's Algorithm – Quantum Complexity Classes and their relationship with classical complexity classes.

Coding Theory Quantum Noise – Introduction to the theory of Quantum Error Correction –Quantum Hamming Bound – Coding Schemes – Calderbank-Shor-Steane codes – Stabilizer Codes.

References

1. M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information,* 1st ed, Cambridge University Press, 2002.
2. J. Gruska, *Quantum Computing*,1st ed, McGraw Hill, 1999.
3. P. R. Halmos, *Finite Dimensional Vector Spaces*, 1st ed, Van Nostrand, 1958.
4. J. Brown, Minds, *Machines and the Multiverse: The Quest for the Quantum Computer*, 1st ed, Simon and Schuster, 2000

CS6122E COMPUTER ARCHITECTURE AND DESIGN

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze different architectural parameters for measuring performance, energy and area.

CO2: Compare different architectures with respect to performance and energy aspects

CO3: Apply software tools for designing various architecture components of a system.

Pipeline and VLIW architecture

Review of pipeline, pipeline hazards,static and dynamic branch prediction, branch target buffer, static scheduling, dynamic scheduling, speculation, superscalar architecture, VLIW architecture.
Introduction to Thread level parallelism, simultaneous multithreading. memory hierarchy design-Cache optimization, RAM.

Thread Level parallelism

Multithreaded processors and Multicore processors: concept, methodologies and analysis, shared memory multiprocessor architecture, cache coherence, coherence protocols, memory consistency, sequential consistency, relaxed consistency models, synchronization, hardware support for synchronization. speculative multithreading, multicore processor design and compilation issues, scheduling,
CMPs and polymorphic processors: concept, studies and analysis, OpenMP programming

Multi-core architecture

Simulators in computer architecture: introduction, methods, ADLs, traces, dynamic compilation, multicore simulators, functional and performance simulators.

References:

1. J. L. Hennessy and D. Patterson, *Computer Architecture: A Quantitative Approach*, 6th ed., Morgan Kaufmann, 2017.
2. ACM SIGARCH *Computer Architecture* News.
3. The WWW *Computer Architecture* page. http://www.cs.wisc.edu/arch/www/ last accessed 23/3/2016.
4. J. P. Shen and M. H. Lipasti, Modern processor design: *Fundamentals of superscalar architectures*, 1st ed., Waveland Pr Inc, 2013.

CS6203E TOPICS IN CRYPTOGRAPHY

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Demonstrate knowledge of historical and modern day ciphers and principles of security behind them.

CO2: Evaluate the security of private key encryption schemes and public key encryption schemes.

CO3: Choose the appropriate encryption scheme for a given context.

Historical Ciphers - Modern day Ciphers. Provable Security and other varying security notions. Perfect Secrecy. One Time Pad. Practical Implementation and limitations. Private-Key Encryption - Constructing Secure Encryption Schemes - Linear and Differential Cryptanalysis. Practical implementation of block ciphers.

Cryptographic Techniques for Integrity - Message Authentication Codes - Hash Functions - Attacks. Practical implementation of hash functions.

Number theory concepts - Primality testing algorithms - Factoring algorithms - Algorithms for computing discrete logarithm. Review of RSA - Attacks on RSA. Introduction to advanced cryptosystems like Elliptic Curve Cryptosystems, Homomorphic Encryption, Threshold Encryption.

References

1. J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, 2nd ed. CRC Press, 2014.
2. H. Delfs and H. Knebl, *Introduction to Cryptography Principles and Applications*, 1st ed. Springer, 2002.
3. D. R. Stinson, *Cryptography Theory and Practice*, 3rd ed. CRC Press, 2006.
4. A. J. Menezes, P. C. Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*,1st ed CRC Press 1996.

CS6204E TOPICS IN DATA PRIVACY

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Identify the data privacy requirements of a system.

CO2: Prepare formal specifications of the privacy requirements.

CO3: Analyze common data privacy techniques and solve simple problems using them.

Introductory Topics

Understanding Privacy - Social Aspects of Privacy, Legal Aspects of Privacy and Privacy Regulations, Effect of Database and Data Mining technologies on privacy, Privacy challenges raised by new emerging technologies such RFID, biometrics, etc.

Privacy Models

Privacy Models - Anonymization models: K-anonymity, l-diversity, t-closeness, differential privacy Database as a service, cloud computing - Data privacy and security law - GDPR

Technology for Preserving Privacy

Using technology for preserving privacy - Statistical Database security, Inference Control, Secure Multi-party computation and Cryptography Privacy-preserving Data mining, Hippocratic databases

Emerging Applications

Emerging Applications - Social Network Privacy, Location Privacy, Query Log Privacy, Biomedical Privacy

References:

1. C. Dwork and A. Roth, *The Algorithmic Foundations of Differential Privacy (Foundations and Trends in Theoretical Computer Science) Vol. 9*, Nos. 3–4 (2014) 211–407, DOI: 10.1561/0400000042
2. N. Venkataramanan and A. Shriram, *Data Privacy: Principles and Practice*, 1st ed, CRC Press, 2016
3. Current Literature based research articles

CS6205E TOPICS IN INFORMATION SECURITY

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Examine and apply the fundamental techniques of information security.

CO2: Demonstrate advanced knowledge in the field of information security standards.

CO3: Detect and analyze diverse kinds of system and network related attacks using software tools.

Information systems - Threats, building blocks of information security, Information security management, information security risk analysis, Vulnerability, Threat and Risk, Risk Assessment and Mitigation; System security topics: Access Control - MAC. DAC, RBAC. Enforcing Access Control: Isolation and Sandboxing, Security Models: BLP, Biba, Chinese Wall and Clark Wilson - Reference Monitor.

Software vulnerabilities - Buffer and stack overflow, Shellcode, Malicious software - Ransomware, Viruses, Rootkits,. Entity Authentication - Password, Challenge Response, Zero Knowledge Protocols.
Network Security Topics - DoS, DDoS, Botnets, iptables/pfSense. DNS security: DNS rebinding attacks, SADDNS, Intrusion Detection Systems - DDoS detection, Malware defense.Web Security: TLS, CSS, XSRF

Current Trends in Information Security, benefits and Issues related to information security. Information security standards": Cobit, Cadbury, ISO 27001, OWASP, OSSTMM, etc. Introduction to BCP / DRP / Incident management, Segregation and Separation of Duties & Roles and responsibilities, IT ACT 2000

References:

1. Nina Godbole, "*Information Systems Security: Security Management, Metrics, Frameworks and Best Practices*", 2nd ed, USA: Wiley, 2017.
2. Michael Goodrich and Roberto Tamassia, "*Introduction to Computer Security*", 1st ed,Pearson,, 2011
3. Ross Anderson, "*Security Engineering: A Guide to Building Dependable Distributed Systems*", 3rd ed, UK: Wiley Publications, 2021.
4. Charles Pfleege and Pearson EduShari Lawrence Pfleeger, "*Security in Computing*",5th ed, USA: Pearson Education; 2018.
5. B. Menezes, *Network Security and Cryptography*,1st ed, India: Cengage Learning, 2010.
6. D. Gollmann, *Computer Security*,3rd ed, John Wiley and Sons Ltd., 2010.
7. Michal Zalewski, "*The Tangled Web: A Guide to Securing Modern Web Applications*", 1st ed USA: No Starch Press, 2011.

CS6206E SYSTEMS SECURITY

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Apply security principles in developing secure systems

CO2: Analyze and interpret the functioning of malicious softwares

CO3: Identify system level and network level attacks and formulate defense strategies

Secure System Design

Secure design principles, Principle of least privileges, information flow control, Linux kernel security, trusted execution environments, secure boot, SELinux

Memory Corruption

Stack Smashing Attacks: Defenses, Stack Smashing Attack techniques, Code Injection and Reuse, Memory Corruption, Heap Overflows, Format-string Attacks, Integer Overflows, Categorization of memory error defenses, Randomization based defenses, memory errors: Definition, Detection, and Prevention, Taint-tracking, Race Conditions, CVE, CWE

Malware analysis

Stealth, Obfuscation, Defenses for Untrusted Code, Reference Monitors, System call interception, process injection, Inline Reference Monitors, Control Flow Integrity, Binary Analysis and Instrumentation, Disassembly and Binary Analysis, Static and Dynamic Binary Translation, APT threats, malware case studies

Network and OS Security

SSL/TLS attack: SSL stripping attack, Heartbleed attack, Beast attack Poodle attack. Processor Vulnerabilities: Meltdown & Spectre, OS Security and Access Control: File Permissions and ACLs, OS Capabilities, Mandatory Access Control, Domain and Type enforcement, Linux Capabilities, Policies for Untrusted. Code, Policy Management. Virtual Machines, Isolation and Sandboxing

References:

1. Stamp,and Mark. *Information security: principles and practice*. 2nd ed, USA: John Wiley & Sons, 2011.
2. Anderson, Ross. *Security engineering: a guide to building dependable distributed systems*. USA: John Wiley & Sons, 3rd ed., 2020.
3. Saltzer, Jerome H., and M. Frans Kaashoek, "*Principles of Computer System Design: An Introduction*", USA: Morgan Kaufmann Publishers, 1st ed, 2009.
4. Christoph Kern, Anita Kesavan, and Neil Daswani, "*Foundations of Security: What Every Programmer Needs to Know*", USA: APress, 1st ed., 2007.

CS6320E STATISTICAL FOUNDATIONS OF DATA SCIENCE

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Construct a probability model for a given data science problem specification and perform computations of probability values and statistical estimates associated with events and random variables.

CO2: Perform modeling of data science problems and computation with random vectors of multivariate Gaussian distribution.

CO3: Apply software packages for performing statistical computations of simple practical problems.

Review of elementary probability: probability spaces, events, independence, conditional probability, Bayes theorem. MAP estimation using Bayesian statistics.

Random variables: expectation, variance and moments, density and distribution functions, weak law of large numbers, functions of random variables. modeling data science problems with probability models.

Jointly distributed random variables, correlation and covariance, joint density and joint distribution, conditional expectation, conditional density and distribution, multivariable Gaussian distribution and its properties.

Transform methods: moment generating functions and characteristic function, Central limit theorem (proof not expected).

Multivariate Gaussian models for data science problems and programming exercises.

Random vectors, properties of covariance matrices, diagonalization, properties of multivariate normal density function, parameter estimation, Sampling theorem and Bessel's correction, estimation of vector means and covariance matrices, maximum likelihood estimation, estimation of multivariate Gaussian, applications to data science problems, use of software packages for multivariate estimation.

Estimation of random variables, minimum mean squared error estimation, mean squared error minimization of Gaussian random vectors, orthogonality equations, Gauss Markov theorem, linear estimation using Gauss Markov statistics.

Introduction to MAP estimation, statistical Learning and PAC learnability, use of software packages.

References

1. A.O. Allen, *Probability, Statistics and Queuing Theory with Computer Science Applications*, 2nd ed, Academic Press, 1990.
2. H. Stark, J.W. Woods, *Probability, Random Processes and Estimation Theory for Engineers*, 2nd ed., Prentice Hall, 1999.
3. C. R. Rao, *Linear Statistical Inference and its applications, John Wiley and sons* 2nd ed, Wiley India, 2009.

CS6305E ADVANCED DATA STRUCTURES AND ALGORITHMS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions:39

Course Outcomes:

CO1: Apply data structures to improve the running time of basic algorithmic questions involving data represented as graphs, lists and other data structures.

CO2: Apply probabilistic and hashing techniques for solving big-data problems and

CO2: Analyze the performance of big-data algorithms for clustering, and graph problems.

Review of Elementary Data Structures and Algorithms

Elementary Data Structures and Algorithms: Time and space complexity analysis, proof of correctness of algorithms,searching and sorting, insertion and selection sorting, divide and conquer algorithms, quick sort, abstract data types, stack, queue, priority queues binary search trees and hashing, algorithms on graphs, spanning trees and shortest path problems.

Hashing for Big Data

Basic probability - random variables, expectation; Standard probability distributions and their properties - Binomial, geometric, Gaussian; Concentration inequalities and applications

Hashing - chaining, open addressing; k-wise independent hash families and their constructions; Perfect hash families - FKS hashing; Bloom filters and applications; Cuckoo hashing; Locality-sensitive hashing and nearest neighbors

Data Streams and Clustering

Data stream algorithms: sampling, counting, heavy hitters with the count-min sketch, Flajolet-Martin algorithm, AMS sketch. Graph streams - connectivity, spanning trees, spanners, sparsifiers

Clustering algorithms: hierarchical clustering, curse of dimensionality, k-means; Clustering in large graphs - random walks and applications; PageRank algorithm

References

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms* 3rd ed. US: MIT Press, 2009.
2. J. Kleinberg and E. Tardos, *Algorithm Design*, 1st ed. Pearson Education, 2013.
3. D. Kozen, *The Design and Analysis of Algorithms*, 1st ed. Springer, 1991.
4. J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of large data sets,* 3rd ed. Cambridge University Press, 2020.
5. E. Upfal and M. Mitzenmacher, *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis,* 2nd ed. Cambridge University Press, 2017.

CS6306E ADVANCED DEEP LEARNING AND COMPUTER VISION

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Illustrate the fundamental concepts of computer vision systems.

CO2: Implement various deep learning architectures used for computer vision.

CO3: Design Deep Learning architectures for computer vision applications.

Course Overview and Motivation; Computer Vision Basics : Introduction to Image Formation, Capture and Representation; Linear Filtering, Correlation, Convolution,Edge, Blobs, Corner Detection; Color, Texture, Segmentation, Scale Space and Scale Selection; SIFT, SURF; HoG, LBP, Bag-of-words, VLAD; RANSAC, Hough transform; Pyramid Matching; Optical Flow, Object Recognition.

Review of Deep Learning, Multi-layer Perceptrons, Backpropagation,Introduction to CNNs; Evolution of CNN Architectures: AlexNet, ZFNet, VGG, InceptionNets, ResNets, DenseNets, Visualization of Kernels; Backprop-to-image/Deconvolution Methods; Deep Dream, Hallucination, Neural Style Transfer; CAM,Grad-CAM, Grad-CAM++; Recent Methods (IG, Segment-IG, SmoothGrad), CNNs for Recognition and Verification (Siamese Networks, Triplet Loss, Contrastive Loss, Ranking Loss); CNNs for Detection: Background of Object Detection, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD, RetinaNet; CNNs for Segmentation: FCN, SegNet, U-Net, Mask-RCNN

Review of RNNs; CNN + RNN Models for Video Understanding: Spatio-temporal Models, Action/Activity Recognition, Introduction to Attention Models in Vision; Vision and Language: Image Captioning, Visual QA, Visual Dialog; Spatial Transformers; Transformer Networks, Transformer model: Introduction of Attention Mechanism, Queries, Keys, and Values of Attention Network, Self-Attention and Positional Encodings, Attention-Based Sequence Encoder, Coupling the Sequence Encoder and Decoder, Cross Attention in the Sequence-to-Sequence Model, Multi-Head Attention, The Complete Transformer Network, BERT based models, Nemo, Self-supervision techniques, masked language modelling, autoregressive modelling.

Review of Deep Generative Models: GANs, VAEs; Other Generative Models: PixelRNNs, ADE,Normalizing Flows, Applications: Image Editing, Inpainting, Superresolution, 3D Object Generation, Security; Variants: CycleGANs, Progressive GANs, StackGANs, Pix2Pix

References:

1. I. Goodfellow, Yoshua Bengio and Aaron Courville, *Deep Learning*,1st ed. MIT Press, 2016
2. M. Nielsen, *Neural Networks and Deep Learning*,1st ed. O'Reilly Media, Inc. 2016
3. Y. Bengio, *Learning Deep Architectures for AI, (Foundations and Trends in Machine Learning),* 1st ed. New Publishers, 2009
4. R. Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed. 2021.
5. S. Prince, *Computer Vision: Models, Learning, and Inference*, 1st ed, Cambridge University Press, 2012.
6. D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, 2nd ed, Pearson, 2012.

CS6307E AI IN HEALTHCARE

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Identify problems healthcare providers face that deep learning can solve, to bring AI technologies into the clinic, safely and ethically.

CO2: Apply the building blocks of AI to understand emerging technologies and to determine the most effective treatments.

CO3: Illustrate the current applications of AI in Radiotherapy, Cardiology, Gynecology.

Neural Networks, Convolutional Neural Networks, Transfer Learning, Data Augmentation, Recurrent Neural Networks, Generative Adversarial Networks (GAN), Autoencoder, VAE, DAE, Segmentation architectures – Unet, Vnet, Object detection DL architectures, Faster RCNN, Mask; RCNN, YOLO, efficient net, centrenet, Transformers, Reinforcement Learning, Sequence model for voice data.

AI in Radiotherapy - Radiotherapy treatment workflow for cancer treatment, Imaging modality for radiotherapy, Computerized Tomography, Fan Beam CT, Cone Beam CT, DICOM file structure, Removing Metal Artifices in CT images using GAN, Treatment Planning System(TPS) in radiation therapy, Atlas based segmentation, Deep Learning based segmentation, segmentation organs in CT images using UNet, Synthesising FBCT images from MR and CBCT images, Linear accelerators, Simulating the operation of a linear accelerator.

AI in Cardiology - processing & analyzing an echocardiogram for reducing intra observer variability, Deep learning based ultrasound processing.
AI in Cervical cancer detection, Pap smear test, VIA & HPV test. Object detection for Cropping RoI in medical images, use of self- supervision in biomedical imaging applications, Development of AI based biomedical devices for cancer detection, Mammographic analysis.

AI and NLP for biomedical applications. Text simplification for deaf learners. Text summarization and Text Simplification using Transformers, Lexical simplification, Syntactic simplification, development of level appropriate robot assistant Chabot for deaf learners, Large Language Models, Chat GPT.
AI and Healthcare: Smart hospitals, pertained networks, Nvidia CLARA framework for healthcare applications like Cancer detection, CT synthesis, segmenting the organs in a CT image

References:

1. Y. Bengio, I. Goodfellow and A. Courville, *"Deep Learning"*,1st ed, MIT Press, 2016.
2. Raúl Rojas, *Neural Networks: A Systematic Introduction*,1st ed, 1996
3. Geoffrey E. *Hinton, "Neural network architectures for artificial intelligence",*1st ed, American Association for Artificial Intelligence Menlo Park, 1988, ISBN:0-929280-15-6.
4. Adrian Rosebrock, *"Deep Learning for Computer Vision with Python", E-Book,* 1st ed, September 2017.
5. Francois Chollet, *"Deep Learning with Python", Manning Publications;* 1st ed,2017, ISBN-10:9781617294433.

CS6308E TOPICS IN APPROXIMATION ALGORITHMS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Identify the design technique used in a given approximation algorithm.

CO2: Design and analyze approximation algorithms for NP hard problems using various combinatorial techniques, randomization and LP methods.

CO3: Apply reduction techniques of proving hardness of approximation of simple combinatorial problems.

Combinatorial Methods I

Review of NP Hardness and reductions, approximation factor, technique of lower bounding optimum, simple approximation algorithms for Vertex Cover and Set Cover. Local search: Max-Cut, Minimum Spanning Tree heuristic, Metric Steiner Tree and Metric TSP, Approximation factor preserving reduction for Steiner tree.

Combinatorial Methods II

Layering: minimum vertex cover, feedback vertex set, Strong NP Hardness and pseudo-polynomial-time algorithms, Knapsack. Dynamic programming: FPTAS for the Knapsack, asymptotic PTAS for Bin Packing.

Linear Programming Methods

Linear programming: simple rounding - MAX-SAT with small clauses, Set Cover, Bin Packing, primal-dual method - set Cover, steiner forest. Introduction to Semidefinite Programming: SDP approximation algorithm for Max Cut.

Non Approximability

Hardness of Approximation: Reductions from NP-complete problems, TSP, approximation preserving reductions, PCP Theorem (no proof), gap reductions, minimum vertex cover, maximum Clique.

References

1. V. V. Vazirani, *Approximation Algorithms*, 1st ed, Springer-Verlag Berlin Heidelberg, 2003.
2. D. P. Williamson and D. B. Shmoys, *The Design of Approximation Algorithms*, 1st ed, Cambridge University Press, 2011.
3. D. S. Hochbaum (Ed), *Approximation Algorithms for NP-Hard Problems*, 1st ed, PWS Publishing Company, 1997.

CS6309E COMPUTATIONAL LINEAR ALGEBRA

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Apply mathematical techniques like singular value decomposition to solve simple engineering applications like least squares approximation or low rank approximation of matrices.

CO2: Prove the correctness of the matrix methods used in engineering problem solving.

CO3: Design computer programs to use the mathematical techniques for solving application problems like image compression or least squares fitting.

Vector spaces, subspaces, linear independence, basis, dimension, inner products, orthogonality, inner product spaces, orthonormal basis, Parsavel's identity, orthogonal subspace, Gram Schmidt orthogonalization. Linear transformations, basis transformations, rank nullity theorem, solving systems of linear equations. Gaussian elimination, LU decomposition. Programming exercises with Linear equations and LU decomposition.

Eigenvalues and Eigenvectors, orthonormal basis transformations, Hermitian operators, projection operators, orthogonal projections, Spectral theorem for Hermitian operators, matrix norms, Frobenius inner product. Symmetric Positive semidefinite matrices and properties, Cholesky decomposition. Programming exercises with Cholesky decomposition.

Singular Value decomposition, Eckart Young theorem, low rank approximation, applications of singular value decomposition, Image compression, least squares approximation, clustering, principal component analysis, pseudo-inverse and properties, Programming exercises with Singular value decomposition, least square approximation and principal component analysis - solution to big-data problems using principal component analysis.

Eigenvalue problems, Gershgorin's theorem, Rayleigh principle, Courant–Fischer min-max principle. programming exercises.

References

1. W. Hodger, *Numerical Linear Algebra - An Introduction, Cambridge Texts in Applied Mathematics,* 1st ed. Cambridge University Press, 2018.
2. D. S. Watkins, *Fundamentals of matrix computations, Pure and Applied Mathematics (Hoboken),* 1st ed. John Wiley & Sons, 2010.
3. B. N. Datta, *Numerical Linear Algebra and Applications*, 1st ed. Society for Industrial and Applied Mathematics (SIAM), 2010.

CS6310E COMPUTATIONAL OPTIMIZATION METHODS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze and solve a linear and non-linear system of equations and modeling of real world engineering problems as an optimization problem.

CO2. Apply nature-inspired optimization algorithms to solve real world optimization problems.

CO3. Design and analysis of computational algorithms dealing with multiple objectives.

Ingredients of Optimization Problems: objective function, constraints, variable bounds. Linear Programming: standard form of linear programming problem, canonical form, elementary operations, graphical methods, simplex method, dual simplex method, linear regression.

Beam Search, Hill Climbing, Tabu Search, Goal Programming, Gradient Descent methods, Simulated Annealing, Fuzzy Optimization.

Nature Inspired Optimization Algorithms, Introduction to Meta-heuristic Optimization, Genetic Algorithms, Differential Evolution, Bee Colony Optimization, Particle Swarm Optimization, Ant Colony Optimization, Working Principles,Applications of Heuristic and Meta-heuristic Optimization in Data Analytics.

Multi-objective Optimization Problems: linear and nonlinear, convex and non-convex optimization, principles of multi-objective optimization, dominance and pareto-optimality, classical methods: weighted sum methods, weighted matrix methods, evolutionary algorithms for optimization: NSGA-II, MOPSO.

References

1. D. G. Luenberger and Yinyu Ye, *Linear and Nonlinear Programming*, 1st ed. Springer, 2016
2. A. P. Engelbrecht, *Computational Intelligence An Introduction*, 1st ed. John Wiley & Sons Ltd, 2007
3. K. Deb, *Multi-objective Optimization methods*, 1st ed. John Wiley & Sons, 2001

CS6311E TOPICS IN DATA MINING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Illustrate the basics of data preprocessing and apply them to clean a data set

CO2: Analyze various classification techniques like - Decision Trees, Rule Based method, Nearest neighbor, Support Vector Machine - and implement them on sample data sets for performance evaluation.

CO3: Implement Apriori and FP-tree methods for frequent item set generation and association rule generation with proper evaluation of the rules.

CO4: Apply and evaluate the performance of various clustering techniques like K-means, DBSCAN and Hierarchical methods on sample data sets

Introduction to data mining-challenges and tasks, Data preprocessing data analysis, measures of similarity and dissimilarity, Data visualization –concepts and techniques

Classification- decision tree-performance evaluation of the classifier, comparison of different classifiers, Rule based classifier, Nearest-neighbor classifiers-Bayesian classifiers-support vector machines, Class imbalance problem

Association analysis –frequent item generation rule generation, evaluation of association patterns

Clustering - Types of clustering, Partition based, Hierarchical, Density based - Cluster evaluation. Application of data mining techniques to various domains

References

1. P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*,1st ed. Pearson Education 2006.
2. J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. Morgan Kaufmann, 2005.
3. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning - Data Mining, Inference, and Prediction*, 2nd ed. Springer, California, 2008.

CS6141E DISTRIBUTED COMPUTING AND BIG DATA

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze a computational problem which is distributed in nature and design approaches for solving it

CO2: Design distributed algorithms for asynchronous distributed computing systems

CO3: Design and implement very large scale distributed databases which support big data processing

Introduction to Computing

Introduction, Parallel Vs Distributed Systems, Characteristics of Distributed Systems, Models of Computation, Happened_before and Causality Relations, Asynchronous Distributed Systems.

Clocks and Distributed Mutual Exclusion Algorithms

Clocks (physical, logical, vector, chain), Verifying Clock Algorithms. Clocks of Different Dimensions, Mutual Exclusion & Distributed Coordination, Using Timestamps - Lamport's Algorithm,Token based and quorum based mutual Exclusion. Topology specific and Topology independent DME algorithms. Drinking Philosophers Problem., Dining Philosophers problem,

Leader Election, Global State and Distributed Consensus

Snapshot algorithms, Spanning tree, Global state Detection algorithms Leader election algorithms,Distributed Consensus/agreement algorithms, Self State Stabilization algorithms, Termination Detection algorithms

Distributed Databases and Big Data

Distributed Databases, Transactions and Concurrency control, Distributed Transactions,Distributed Recovery, Checkpointing for Recovery, Message Logging for Recovery, Big Data Processing, Introduction to Cloud Computing and virtualization.

References:

1. Vijay K. Garg., *Elements of Distributed Computing*,1st ed, Wiley & Sons, 2020
2. Vijay K. Garg., *Concurrent and Distributed Computing in Java*, 1st ed, Wiley & Sons, 2020
3. Ajay D. Kshemkalyani, Mukesh Singhal,*Distributed Computing Principles, Algorithms, and Systems* , Cambridge University Press, 2021
4. Chow R. & Johnson T., *Distributed Operating Systems and Algorithms*, Addison Wesley, 2022
5. Tanenbaum S., *Distributed Operating Systems*, Pearson Education, 2021

CS6313E HIGH PERFORMANCE COMPUTING FOR AI

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Illustrate the neural network basics and analyze commonly used networks for different AI needs.

CO2: Analyze and compare parallel computing paradigms.

CO3: Design GPU/FPGA programs for Deep Learning frameworks.

Introduction to Artificial Intelligence, AI workloads: Neural Networks ANN, Convolutions, CNN, Image Classification, Object detection, Region Proposal Networks, Faster RCNN, Yolo, Image segmentation, UNET, UNET++, Generative Adversarial Networks, Generators, Discriminators, Reinforcement Learning Models, RNNs, LSTMs, Transformers, Importance of matrix operations and the need of parallel kernels in Deep learning, Deep Learning Hardware and Software.

Introduction to Parallel Programming; Needs for parallel computations. Homogenous parallel computing; programming models (CPU only Models), Programming with Shared Memory; Overview of the Openmp standard, Overview of the MPI standard. Point-to-point communication operations. Synchronous and asynchronous modes of data transmission. Case studies: matrix computations, solving partial differential equations using OpenMP and MPI.

Heterogeneous parallel computing; Accelerators, GPUs, CUDA, Overview of CUDA C; threads, blocks and grids, warps, different GPU memories, CUDA Kernels, Operations in Deep Learning and their implementation on CUDA. Designing High Performance Systems for Accelerated Machine Learning and Deep Learning Workloads. Deep Learning Software: Setting up Application Environment for Deep Learning and HPC workloads using Container Platform like Docker. Introduction to PyTorch, TensorFlow (Majorly Used).

Deep learning on FPGA: Introduction to FPGAs, Architecture of FPGAs, Implementation of complex digital computations with FPGAs, FPGAs for AI, the challenges of using GPUs for deep learning, FPGAs vs. GPUs for Deep Learning, Different deep learning architectures for FPGAs, Deep Learning Accelerator scaling on FPGA Creating ASICs for AI. Pros and cons of using FPGAs for AI workload acceleration.

References:

1. Georg Hager and *Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers*, 1st ed. Chapman & Hall / CRC Computational Science series, 2011.
2. David Kirk Wen-mei Hwu, *Programming Massively Parallel Processors, A Hands-on Approach*, 2nd ed. Morgan Kaufmann
3. Jason sanders and Edward Kandrot, *CUDA by Example: An Introduction to General-Purpose GPU Programming*, 1st ed. Addison-Wesley
4. Michael J. Quinn, *Parallel Programming in C with MPI and OpenMP*, 1st ed. McGraw-Hill
5. Palnitkar S. *Verilog HDL: a guide to digital design and synthesis*. 1st ed. Prentice Hall Professional; 2003.Y. Bengio, I. Goodfellow and A. Courville, “Deep Learning”, MIT Press, 2016.
6. Wolf W. *FPGA-based system design*. 1st ed. Pearson Education India; 2004.
7. Kilts S. *Advanced FPGA design: architecture, implementation, and optimization*. 1st ed. John Wiley & Sons; 2007 Jun 18.
8. Adrian Rosebrock, “*Deep Learning for Computer Vision with Python*”, E-Book, 1st ed. September 2017.

CS6314E INTELLIGENT AGENTS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Illustrate the fundamental concepts and functions of intelligent agents.

CO2: Analyze the methodologies for agent interaction negotiation and decision making

CO3: Apply the principles and methods of intelligent agents to practical problems

Introduction to agents and agent based systems

Motivations for agent-based computing, abstract architecture of agents, types of agents, concepts and models of reasoning, rational decision making and handling uncertainty.

Multi agent systems

Classifying multi-agent interactions - cooperative versus non-cooperative agents, models of negotiation, cooperation, communications and reaching agreement in multiagent systems Interaction languages and protocols.

Agent design and implementation

Application of intelligent agents in complex distributed problem solving, deploying agents within a simulated environment, practical reasoning strategies for distributed scenarios.

References:

1. M. Wooldridge, *An Introduction to MultiAgent Systems* , 2nd ed. Wiley, 2009.
2. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach* , 3rd ed. Prentice Hall, 2009.
3. Current Literature.

CS6315E: INTERNET OF THINGS

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: To illustrate the technology and standards relating to IoT

CO2: To design and develop domain specific IoT systems

CO3: To analyze the IoT applications and provide domain specific solutions

Introduction to Internet of Things

Introduction to IoT, things in IoT, IoT functional blocks, IoT communication models and communication APIs, sensors, actuators, smart objects, basics of sensor networks, basics of networking and communication protocols, machine-to-machine communications, IoT network architecture & design, relationship of IoT with SDN and NFV, cloud, and fog computing

Device and Communication layer

IoT access technologies, physical layer and MAC layer, LoRaWAN, IEEE 802.11ah and IEEE 805.12.4, IoT network layer: need for optimization, optimizing IP for IoT - 6LoWPAN to 6Lo, 6TiSCH and RPL

Application layer

IoT application layer: IoT application and transport methods, CoAP, AMQP, message queues and publish/subscribe model, MQTT, interoperability – MQTT vs. AMQP

Domain specific IoT

Domain specific IoTs: smart lighting, smart appliances, structure healthcare monitoring, pollution monitoring, surveillance. IoT physical devices and endpoints, servers and cloud platform, data analytics for IoT - machine learning overview, case studies on illustrating IoT design

References

1. Adrian McEwen and Hakim Cassimally, "*Designing the Internet of Things*", UK: Wiley Publications, 1st ed. 2014
2. Jean-Philippe Vasseur and Adam Dunkels, "*Interconnecting Smart Objects with IP: The Next Internet*", 1st ed, USA: Morgan Kaufmann Publications, 2010
3. Edward Ashford Lee and Sanjit A Seshia,"*Introduction to Embedded Systems –A Cyber–Physical Systems Approach*", 2nd ed. USA: MIT Press; 2017
4. Hanes David, Salgueiro Gonzalo, Grossetete Patrick, Barton Rob, and Henry Jerome, "*IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things*", 1st ed, UK: Pearson Education, 2017
5. Simone Cirani, Gianluigi Ferrari, Marco Picone, and Luca Veltri, "*Internet of Things: Architectures, Protocols and Standards*", 1st ed, USA: Wiley–Blackwell, 2018
6. Recent peer reviewed journal papers

CS6316E MUSIC INFORMATION RETRIEVAL

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: To illustrate fundamental concepts of music information retrieval.

CO2: To perform audio feature extraction and analysis of music.

CO3: To design and implement solutions for a given music information retrieval problem.

Extracting Information From Music Signals

Introduction to music Information retrieval (MIR): history and evolution.
Music modalities and representations: time, frequency, and sinusoids, DFT and time-frequency representations, audio feature extraction

Fundamental MIR tasks

Pitch Detection: rhythm analysis, genre classification, emotion recognition, auto-tagging, music similarity, evaluation in music information retrieval.

Advanced Tasks and Retrieval Systems

Query Retrieval, structure segmentation, chord detection and cover song identification, music recommendation systems

References:

1. M. Müller, *Fundamentals of music processing: Audio, analysis, algorithms*, applications. Vol. 5. Cham: Springer, 2015.
2. T. Li, M. Ogihara, and G.Tzanetakis, Eds, *Music data mining*, Vol. 20., Boca Raton: CRC Press, 2012.

CS6317E TOPICS IN NATURAL LANGUAGE PROCESSING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Analyze the applications of language modeling, information extraction, named entity recognition, information retrieval, text classification, word sense disambiguation, automatic question answering and text summarization in real world problems.

CO2: Illustrate the complexity of natural language grammatical structures with an emphasis on the English language by demonstrating various parsing techniques

.

CO3: Apply natural language processing (NLP) tools and libraries (such as python, nltk) and develop software tools for various NLP tasks such as tagging, parsing and text classification.

CO4: Design new solutions to issues of current interest from recently published literature, and do a comparative analysis of different possibilities.

Natural Language Understanding

Natural Language understanding: The study of language, Applications of NLP, Evaluating language understanding systems, Levels of language analysis, Representations and Understanding, Organization of Natural Language Understanding systems, Linguistic background: An outline of English syntax.

Grammars and Parsing

Grammars and parsing: grammars and sentence structure, top-down and bottom-up parsers, transition network grammars, top-down chart parsing. feature systems and augmented grammars: basic feature system for English, morphological analysis and the lexicon, parsing with features, Augmented Transition Networks**.**

Grammars for Natural Languages

Grammars for Natural language: Auxiliary Verbs and Verb Phrases, Movement phenomenon in language, Handling questions in context-free grammars,Human preferences in parsing, Encoding uncertainty, Deterministic parser, Ambiguity resolution: Statistical methods, Estimating probabilities.

Advanced Topics

Part-of-Speech tagging, Probabilistic Context-free Grammars, Semantics and Logical form, Word senses and Ambiguity, Information Extraction, Named Entity Recognition, Machine Translation, Summarization, Question Answering, Recent trends in NLP - LLMs

References

1. J. Allen, *Natural Language Understanding*, 2nd ed, Pearson Education, 2003.
2. D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 2nd ed, Pearson Education, 2009.
3. C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, 1st ed, The MIT Press, Cambridge, Massachusetts.1999.
4. Current Literature.

CS6318E NEURAL NETWORKS AND DEEP LEARNING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Define basics of neural networks and learning rules of deep neural networks.

CO2: Analyze the mathematical, statistical and computational challenges of building stable representations for high-dimensional data, such as images and text using case studies and propose appropriate solutions.

CO3: Apply deep learning algorithms to solve simple real world problem situations.

Biologically inspired computing, historical context, Perceptron Learning rule, Backpropagation, Multi-layer Perceptron model, Activation Functions – Sigmoid, Tanh, ReLU, Leaky ReLu, Loss functions, Optimization: Stochastic gradient descent, Training Neural Networks, weight initialization, batch normalization, hyper parameter optimization, parameter updates, model ensembles, dropout, Variance, Bias.

Convolutional Neural Networks: introduction, history, architectures, convolution layer, pooling layer, fully connected layer, Conv Net, Case study of ImageNet challenge -LeNet, AlexNet, VGG, GoogLeNet, ResNet, InceptionNet, EfficientNet etc. Regularization Techniques, Data Augmentation – zooming, rotation, cropping, blurring, noise addition, self-supervision techniques, Transfer Learning, freezing the input layers, fine tuning output layers.

Image Localisation, Image segmentation, masks, Image segmentation architectures – Unet, VNet, UNet++, Object Detection – Region Proposal Networks, Objection architectures RCNN, Fast and Faster RCNNs, Mask RCNN, YOLO, BiFPN layers, Centre Net, EfficientDet, Case study – RoI cropping in CT images and Cervical Images.

Sequential models, Recurrent Neural Networks, Long Short Term Memory, Gated Recurrent Units, NLP based Applications, Identifying missing words in a paragraph, text summarization
Deep Learning Hardware and Software, CPUs, GPUs, GPU architectures – Pascal, Volta, Turing & Ampere, Data Parallelism in GPU, Kernels – vector addition, vector multiplication, matrix addition, matrix multiplication, TPUs, Frameworks for Deep Learning - PyTorch, TensorFlow, Keras, Theano, Caffe, Nvidia DGX machines for DL applications.

References

1. Y. Bengio, I. Goodfellow and A. Courville, *Deep Learning*, 1st ed MIT Press, 2016.
2. Bishop C. M., *Pattern Recognition and Machine Learning*, 1st ed Springer, 2006.
3. Geoffrey E. Hinton, *Neural network architectures for artificial intelligence*,1st ed American Association for Artificial Intelligence Menlo Park, 1988, ISBN:0-929280-15-6.
4. Adrian Rosebrock, *Deep Learning for Computer Vision with Python*, E-Book, 1st ed, September 2017.

CS6319E SPEECH INFORMATION PROCESSING

Pre-requisites: NIL

L	T	P	O	C
3	0	0	6	3

Total Lecture Sessions: 39

Course Outcomes:

CO1: Illustrate the fundamental concepts of Speech Processing.

CO2: Apply various Speech analysis & Modeling techniques used for building speech processing modules.

CO3: Design and build speech recognition systems using Deep Learning architectures.

Basic Concepts

Speech Fundamentals: Articulatory Phonetics – Production and Classification of Speech Sounds; Acoustic Phonetics – acoustics of speech production; Review of Digital Signal Processing concepts; Short-Time Fourier Transform, Filter-Bank and LPC Methods.

Speech Analysis

Speech Analysis: Features, Feature Extraction and Pattern Comparison Techniques: Speech distortion measures – mathematical and perceptual – Log Spectral Distance, Cepstral Distances, Weighted Cepstral Distances and Filtering, Likelihood Distortions, Spectral Distortion using a Warped Frequency Scale, LPC, PLP and MFCC Coefficients, Time Alignment and Normalization – Dynamic Time Warping, Multiple Time – Alignment Paths.

Speech Modeling

Speech Modeling: Hidden Markov Models: Markov Processes, HMMs – Evaluation, Optimal State Sequence – Viterbi Search, Baum-Welch Parameter Re-estimation, Implementation issues.Speech Recognition: Large Vocabulary Continuous Speech Recognition: Architecture of a large vocabulary continuous speech recognition system – acoustics and language models – ngrams, context dependent sub-word units;

Applications and Advanced Topics

Applications and present status. Speech Synthesis: Text-to-Speech Synthesis: Concatenative and waveform synthesis methods, subword units for TTS, intelligibility and naturalness – role of prosody, Applications and present status.

Introduction Speech Processing using Deep learning, Recurrent neural networks, parameter learning with backpropagation, vanishing and exploding gradients, Introduction to long short term memory (LSTM) networks. Introduction to convolutional neural networks.

References

1. Lawrence Rabiner and Biing-Hwang Juang, "*Fundamentals of Speech Recognition*", 1st ed. Pearson Education.
2. Daniel Jurafsky and James H Martin, "*Speech and Language Processing – An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*",1st ed. Pearson Education.
3. Thomas F Quatieri, "*Discrete-Time Speech Signal Processing – Principles and Practice*",1st ed. Pearson Education.
4. Frederick Jelinek, "*Statistical Methods of Speech Recognition*",1*st* ed. MIT Press.
5. Dong Yu, Li Deng, "*Automatic Speech Recognition, A Deep Learning Approach*",1st ed. Springer, 2014.