Sidho-Kanho-Birsha University, Purulia

Ranchi Road, PO-Sainik School Purulia

Dist- Purulia, West Bengal, India

PIN-723104

www.skbu.ac.in

Department of Computer Science

Syllabus for M.Sc. in Computer Science

with effect from the

Academic Session 2020-21

Syllabus of Post Graduate Course in Computer Science(ChoiceBased CreditSystem)

SYLLABUSOVERVIEW

P.G. course of studies in Computer Science shall be of two years duration divided into four Semesters:Semester I, II, III and IV each of six months duration and the Term End Examinations will beconducted at the end of each semester. Syllabus for P.G. course in Computer Science is hereby framedbasedon thefollowingschemes and structures.

Using the guidelines of the University, Choice Based Credit System is offered. Core Courses shouldcompulsorily be studied by all the students of the department. The major elective courses will run inSemester III and in Semester IV. For major elective courses a student would choose from a pool ofcourses (offered by the department divided in groups) from his/her main subject of study. In case ofopen elective course, a student would choose from a pool of courses from other department(s). TheOutreach Programme is one kind of extension activity towards the society which will be helpful for astudent in skill development and direct communication to the society. For Add-on Course, the studentshave tochoosea coursefromapool ofcourses.

Total marks are 1200 of which each semester is of 300 marks. 20% marks are allotted for InternalAssessment for each theoretical paper. There is only one practical paper which is in Semester II. InSemester III, the first two theoretical papers are general and there are also two special papers, the lastbutonetheoreticalpaper isopenelectivepaperandthelastoneis“OutreachProgramme”.InSemester IV, thefirst two theoretical papers are general and the next two papers arespecial papers,the fifth paper is the Add on Course. The project work/dissertation paper is last paper in this semesterand the marks distribution for this is as follows: 20 Marks for written submission, 20 Marks forSeminar presentation and 10 Marks for Viva-Voce. Faculty members of the department will supervisethe students for project work**.** In Semester III the department will offer a cluster of special papersdivided into groups and the students will have to choose special papers according to the norms to bedecided by the Department.The corresponding papers are to be continued as special papers inSemester IValso.

PROGRAMMEOUTCOMES

Computer Science is becoming a popular choice for students around the world. A PG degree in Computer Science can open several new opportunities, especially getting to work with trending technologies such as Big data, Machine learning, Data handling, Network architects, Database Management. ThePGcourse under CBCS is so designed to make the learners to master in the subject. The learnerscan learn almost all areas of Computer Science and Applications. A number of special papers areoffered to cover the recent research areas so that the students have the chance for research. Thesyllabus is highly oriented to the NET/SET/GATE and other competitive examinations and thelearnerswillbeabletocracktheNationalandInternationallevelofexaminationsaftercompletion of the course. We are offering special papers which have huge industrial, businessandengineeringapplicationsandalsoresearchscopes.Aftercompletion ofthecoursethestudents will not only earn the PG degree but they will be able to crack several examinationslike, SSC, PSC, UPSC, RAIL, etc. The Outreach Programme will help the studentsto understand the outreach peoples to make them understand the usage of Computer Science and Applications. TheAdd-on-Course includes the Machine Learning, Python, R which will be helpful to develop their skills.In the project paper the students will be given some advanced topics as their dissertation papersand they will be oriented for research activities. They will learn the type settings, presentationskill andinteraction methods.

Curriculum Structure

SEM	Paper Code	Paper Title	Credits	Marks*
SEM-I	MCSCCT101	Programming Languages	4	40+10
	MCSCCT102	Advance Operating Systems	4	40+10
	MCSCCT103	Design and Analysis of Algorithms	4	40+10
	MCSCCT104	Mathematical Foundations	4	40+10
	MCSCSP105	Operating System Lab	4	50
	MCSCSP106	Programming & Algorithm Analysis Lab	4	50
SEM-II	MCSCCT201	Formal Languages and Automata Theory	4	40+10
	MCSCCT202	Advanced Database Management	4	40+10
	MCSCCT203	Advanced Computer Networks	4	40+10
	MCSCCT204	Information Security and Coding Theory	4	40+10
	MCSCSP205	Network Lab	4	50
	MCSCSP206	Database Lab	4	50
SEM-III	MCSCCT301	Compiler Design	4	40+10
	MCSCCT302	Artificial Intelligence and Expert Systems	4	40+10
	MCSMET303	Major Elective-I (Special Paper-I)	4	40+10
	MCSCSP304	Compiler Design and Artificial Intelligence Lab	4	40+10
	MCSOET305	Open Elective	4	50
	MCSOPP306	Outreach Programme	4	50
SEM-IV	MCSCCT401	Data Analytics	4	40+10
	MCSCCT402	Advanced Software Design	4	40+10
	MCSMET403	Major Elective-II(Special Paper-II)	4	40+10
	MCSMET404	Major Elective-III(Special Paper-III)	4	40+10
	MCSACT405	Add on Course	4	50
	MCSMEP406	Project and Seminar	4	50

***(Internal Assessment: 10 Marks, Semester End Exam: 40 Marks)**

MAJOR ELECTIVES/ SPECIAL PAPERS

Students must choose the Major Electives I, II and III from the same group

Elective Group-A: Data Science

A1: Machine Learning
A2: Deep Learning
A3: Computer Vision and Pattern Recognition
A4: Business Analytics
A5: Natural Language Processing
A6: Big Data Modelling and Management

Elective Group-B: Cyber Security

B1: Network Security
B2: Digital Forensic
B3: Post Quantum Cryptography
B4: Hardware Security
B5: Cyber Law and Ethics
B6: Information and Coding Theory

Elective Group-C: Distributed System and Resources

C1: Distributed Systems
C2: Internet of Things
C3: Service Oriented Computing
C4: Semantic Web
C5: Multimedia Systems and Virtual Reality
C6: Cloud Computing

Detailed Syllabus

Semester-I

MCSCCT101:Programming Languages

Language Design and Translation Issues: Programming Language Concepts, Paradigms and Models. **(L-5)**

Programming Environments: Binding and scope, Programming, Language Syntax and semantics, Stages in Translation, Formal Transition Models. Compilers, Interpreters, Interactive development tools, Debugging tools, variables, data abstraction. **(L-10)**

Memory management: Static & dynamic allocation, control structures, selective structures. Modular programming, function, parameter passing methods, lifetime of variables, recursion, error handling. **(L-10)**

Elementary Data Types: Properties of Types and Objects; Scalar and Composite Data Types. **(L-5)**

Programming Languages: Object oriented-programming, Event driven programming, Exception handling, Concurrent programming, Foundations of functional programming: λ-calculus, type checking, Logic programming, Scripting languages. **(L-15)**

Object Oriented Programming using python: Characteristics of object-oriented programming, Classes and Objects, Methods. Operator Overloading, super () and Method Overriding, abstraction, inheritance, polymorphism, Design Principles for object-oriented programming. **(L-15)**

References:

1. R. Sebesta, Concepts of Programming Languages, Addison Wesley**.**
2. John C Mitchell, Concepts in Programming Languages, Cambridge University Press, 2003.
3. Ravi Sethi, Programming Languages, Addison Wesley, 1996.
4. Van Roy, Haridi, Concepts, Techniques and Models of Computer Programming, MIT Press, 2004.
5. Rambaugh, James Michael, Blaha – "Object Oriented Modelling and Design" – Prentice Hall, India
6. Bruce Eckel-Thinking in Java, 4thEdn.

MCSCCT102:Advance Operating systems

Distributed Systems: Characteristics of distributed System: Examples of distributed systems issues in the design of distributed system. System models: Architectural models and fundamental models. Distributed objects and remote invocation: communication between distributed objects – remote procedure call – Events and notification. Operating system support: Operating system layer – protection – processes and threads communication and invocation – Operating system architecture security: Overview of security techniques. **(L-16)**

Distributed file system: File service architecture - network file system- Andrew file system-recent advances Transactions and concurrency control: nested transactions-locks-optimistic concurrency control comparison of methods for concurrency control-flat and nested distributed transactions- distributed deadlock transactions recovery. Replication System model and group communication- fault tolerant services-transactions with replicated data. **(L-16)**

Real-Time: Real-time process concepts, categories of real-time task, real-time scheduling. Networks: **(L-12)**

Cloud Computing: Overview, Introduction to service and service-oriented architecture, Layers and types of cloud service model, Uses of Cloud; Components of Cloud Computing - Software as a Service, Platform as a Service, Infrastructure as a Service, Identity as a Service; Data storage and virtualization in the cloud. George Coulouris, Jean Dollimore and Tim Kindberg, Distributed Systems: Concepts and Design - Pearson Education. **(L-16)**

References:

1. Andrew S Tanenbaum and Marten Van Steen, Distributed Systems: Principles and paradigms – Pearson Education.

2. Venkatakrishnaŋl, Principles of Grid Computing – Concepts And Applications, Ane Books.

3. Kris Jamsa, Cloud Computing, Jones &Bartlett Learning.

4. Rahul Deva &GarimaKulshreshtha, Soft Computing, Shrof Publishers & Distributors Pvt.Ltd.

5. Rajkumar Buya and etal, Cloud Computing – Principles and Paradigms, Wiley Publishers.

MCSCCT103:Design and Analysis of Algorithms

Algorithms: Properties of good algorithms, Efficiency of algorithms,Time and Space complexity. Non-linear data structures: trees, Binary search tree- algorithms on BST, balanced trees- AVL rotations, multi-way search trees- B Tree, B+ tree. Basic concepts of Red-Black tree, splay tree, tries. **(L-10)**

String Processing: KMP, Boyre-Moore, Rabin Karp algorithms. **(L-8)**

Introduction to randomized algorithms: Random numbers, randomized quick sort, randomly built binary search tree. **(L-8)**

Number Theoretic Algorithms: GCD, addition and multiplication of two large numbers, polynomial arithmetic, Fast-Fourier transforms. **(L-8)**

Advanced Techniques to analyse algorithms: Use and study advanced data structures union find (Disjoint Set Structure), Fibonacci heaps. **(L-8)**

Graph algorithms: Matching and Flows, Graph capture. **(L-6)**

Parallel algorithms: Basic techniques for sorting, searching and merging in parallel. **(L-6)**

Geometric algorithms: Point location, convex hulls and Voronoi diagrams. **(L-3)**

Complexity Theory: P and NP Class Problems; NP-completeness and Reducibility. **(L-3)**

References:

1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009.
2. Sarabasse& A.V. Gelder, Computer Algorithm – Introduction to Design and Analysis, Publisher,Pearson 3rd Edition 199.
3. Jon Kleinberg and Eva Tardos, Algorithm Design, Pearson New International Edition.
4. Aho, J. Hopcroft and J. Ullman: The Design and Analysis of Computer Algorithms, A. W. L, International Student Edition, Singapore, 1998.
5. S. Baase: Computer Algorithms: Introduction to Design and Analysis, 2nd ed., Addison-Wesley, California, 1988.
6. E. Horowitz and S. Sahni: Fundamental of Computer Algorithms, Galgotia Pub. /Pitman, New Delhi/London, 1987/1978.

MCSCCT104:Mathematical Foundations

Mathematical Logic: Propositional and Predicate Logic, Propositional Equivalences, Normal Forms, Predicates and Quantifiers, Nested Quantifiers, Concepts of descriptive logic. Rules of Inference. **(L-8)**

Probability and Statistics: Discrete probability, simple random variables, the law of large numbers, Binomial, Poisson and normal distributions, central limit theorem, non-uniformprobability distributions, Bayes' Theorem on conditional probability. Importance of random sampling, sampling distributions (Normal, t-distribution, chi-square distribution, F-distribution, ANOVA); statistical inferencing. Rejection method, Metropolis algorithm, random variables, expectations and moments, Stochastic process, Markov model. **(L-12)**

Sets and Relations: Permutation Functions, Growth of Functions. Partially ordered sets, Lattices, Finite Boolean algebra. **(L-6)**

Counting Mathematical Induction: Basics of Counting, Pigeonhole Principle, Permutations and Combinations, Inclusion-Exclusion principle, Pigeonhole Principle, Mathematical Induction. **(L-5)**

Algebraic structures: Semigroups, Monoids, Groups, Subgroups, Symmetric groups, Groups homomorphism and isomorphism, Cosets and Lagrange's Theorem, Normal subgroups, Permutation of groups and Burnsides theorem. Boolean Functions and its Representation, Simplifications of Boolean Functions. **(L-10)**

Graph Theory: Basic Concept of Graph Theory, Euler Paths and Circuits, Hamiltonian Paths and Circuits, Spanning tree. **(L-7)**

Optimization Technics: Linear Programming Problems - Mathematical Model, Graphical Solution, Simplex and Dual Simplex Method, Sensitivity Analysis. Integer Programming, Transportation and Assignment Models. PERT-CPM: Diagram Representation, Critical Path Calculations, Resource Levelling, Cost Consideration in Project Scheduling. Introduction to nonlinear programming problems. **(L-12)**

References:

1. J.P. Tremblay & R. Manohar, Discrete Mathematical Structures with Application to computer science [Tata McGraw –Hill]

2. Bernard Kolmanc,Busby& Sharon Ross, Discrete Mathematical Structures

3. N. L. Biggs: Discrete Mathematics, Oxford Science Publications.

4.W. Feller, W: An Introduction to Probability Theory and its Applications, Vol.1, John Wiley.

5. G. R. Grimmett and D. R. Stirzaker: Probability and Random Processes, Oxford Science Publications.

6. J. Nesetril, J. Matousek: Invitation to Discrete Mathematics, Clarendon Press.

7. Grassmann, Logic and Discrete Mathematics: A Computer Science Perspective, Pearson Education, 2007.

MCSCCP105: Operating System Lab

- Distributed System. **(L-30)**
- Open-source cloud. **(L-30)**
- Multi-threaded programming. **(L-30)**
- Network operating system. **(L-30)**

MCSCCP106: Programming & Algorithm Analysis Lab

Python Programming: Running Python Programs and User Interaction, Variables and Expressions, Data Types in Python, Advanced Data Types (List, Set, Tuples, related operations, Dictionary**. (L-20)**

Conditional expressions and Loops: Conditional Expressions, Loops, Nested Loops, Exception Handling. **(L-30)**

Functions, Recursions: Syntax and Basics of a Function, Use of a function, Parameters and Arguments, Return statement, Local andGlobal Scope of a Variable, Recursive functions. **(L-10)**

File Handling: Need of File Handling, Reading/Writing Text and Numbers to/from a File, Directories on a disk. **(L-30)**

Modules and packages: Understanding and Creating Python modules and packages for modular programming, NumPy, Pandas, MatPlotLib, and NLTK**. (L-30)**

Programming using Python:

1. Object Oriented Programming: Function definitions: pattern matching, induction.
2. Basic data types, tuples, lists.
3. Higher order functions.
4. Polymorphism.
5. Reduction as computation, lazy evaluation.
6. Measuring computational complexity.
7. Basic algorithms: sorting, backtracking, dynamic programming.
8. User-defined datatypes: enumerated, recursive and polymorphic types.
9. Input/output.

Design and Analysis of Algorithms Lab:

1. i. Implement Insertion Sort (The program should report the number of comparisons).

ii. Implement Merge Sort (The program should report the number of comparisons).

2. Implement Heap Sort (The program should report the number of comparisons).

3. Implement Randomized Quick sort (The program should report the number of comparisons).

4. Implement Radix Sort.

5. Create a Red-Black Tree and perform following operations on it: i. Insert a node ii. Delete a node iii. Search for a number & also report the colour of the node containing this number.

6. Write a program to determine the LCS of two given sequences.

7. Implement Breadth-First Search in a graph.

8. Implement Depth-First Search in a graph.

9. Write a program to determine the minimum spanning tree of a graph.

Semester-II

MCSCCT201: Formal Languages and Automata Theory

Mathematical Preliminaries for Finite Automata and Formal Languages: Sets, Relations and Functions, Graphs and Trees, Mathematical Induction, Mathematical Logic, Formal Language, Chomsky Hierarchy Languages (CHL), Automata Theory. **(L-3)**

Regular Expressions and Regular Languages: Regular Expressions, Regular Languages and Regular Grammar**. (L-5)**

Finite State Automata: Deterministic Finite Automata, Non-Deterministic Finite Automata, NFA and Regular Expressions, Conversion of Finite Automata to Regular Expression, Conversion of NFA to DFA, NFA with e-Transitions (e-NFA), Conversion from e-NFA to NFA, Conversion from e-NFA to DFA, Output Associated with Finite Automata, Moore and Mealy Machines, Minimization of Automata**. (L-8)**

Properties of Regular Languages: Regular sets, regular expressions, identity rules, constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets (proofs not required). **(L-8)**

Grammar Formalism: Regular grammars-right linear and left linear grammars, equivalence between regular linear grammar and FA, inter conversion, Context free grammar, derivation trees, sentential forms. Right most and leftmost derivation of strings. **(L-7)**

Context Free Grammars: Ambiguity in context free grammars. Minimisation of Context Free Grammars. Chomsky normal form, Greiback normal form, Pumping Lemma for Context Free Languages. Enumeration of properties of CFL (proofs omitted). **(L-7)**

Push Down Automata: Push down automata, definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, interconversion. (Proofs not required). Introduction to DCFL and DPDA**. (L-7)**

Turing Machines: Turing Machine, definition, model, design of TM, Computable functions, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing machines (proofs not required). **(L-8)**

Computability Theory: Chomsky hierarchy of languages, linear bounded automata and context sensitive language, LR(0) grammar, decidability of, problems, Universal Turing Machine, undecidability of posts. Correspondence problem, Turing reducibility, Definition of P and NP problems, NP complete and NP hard problems**. (L-7)**

References

1. "Introduction to Automata Theory Languages and Computation". Hopcroft H.E. and Ullman J. D.Pearson Education.
2. Introduction to Theory of Computation –Sipser 2nd edition Thomson.
3. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.

4. Introduction to languages and the Theory of Computation,John C Martin, TMH.
5. “Elements of Theory of Computation”, Lewis H.P. &Papadimition C.H. Pearson/PHI.
6. Theory of Computer Science – Automata languages and computation -Mishra andChandrashekaran, 2nd edition, PHI.

MCSCCT202: Advanced Database Management

Database System Concepts and Architecture: Data Models, Schemas, and Instances; Three-Schema Architecture and Data Independence; Database Languages and Interfaces; Centralized and Client/Server Architectures for DBMS**. (L-8)**

Data Modelling: Relational Schemas, Update Operations and Dealing with Constraint Violations; Relational Algebra and Relational Calculus; Codd Rules.
SQL: Views, Stored Procedures and Functions; Database Triggers, SQL Injection. **(L-8)**

Normalization and Transactional Management: Functional Dependencies and Normalization; Algorithms for Query Processing and Optimization; Transaction Processing, Concurrency Control Techniques, Database Recovery Techniques, Object and Object-Relational Databases; Concepts of ACID and CAP theory, Database Security and Authorization. Transactions, Recovery, System Recovery, Media Recovery, Two-phase Commit. **(L-10**)

Enhanced Data Models: Temporal Database Concepts, No SQL databases- document oriented models, key-value models, columnar data models, graph data model, Multimedia Databases, Deductive Databases, link data management. **(L-10)**

Distributed Database: Distributed Database System, Distributed Database Design, Data Fragmentation, Data Replication, Data Allocation, Query Processing in Distributed Databases**. (L-10**)

Data Warehousing: Data Modelling for Data Warehouses, Concept Hierarchy, OLAP and OLTP**. (L-8)**

Big Data Systems: Big Data Characteristics, Types of Big Data, Big Data Architecture, Introduction to Map-Reduce and Hadoop; Distributed File System, HDFS. **(L-6)**

References:

1. Ceri S. Pelagatti. G, Distributed Database systems Principles and Systems, McGraw Hill.
2. Silberschatz, Korth, Database System Concepts, TMH.
3. Elmasri&Navathe, Fundamental of Data Base System, The Benjamin Cummins Publishing Inc.
4. Silberschatz, Korth and Sudarshan, Database System Concepts – 6th Edition.
5. C.J. Date, Database Design and Relational Theory: Normal Forms and All That Jazz.

MCSCCT203: Advanced Computer Networks

Data Communication: Components of a Data Communication System, Digital and Analog Transmission; Data Encoding and Modulation Techniques; Broadband and Baseband Transmission; Multiplexing, Transmission Media, Transmission Errors, Error Handling Mechanisms. **(L-8**)

Network Models: Layered Architectures Reference Model and its Protocols; TCP/IP Protocol Suite, Physical, Logical, Port and Specific Addresses; Switching Techniques. **(L-8)**

Functions of OSI and TCP/IP Layers: Framing, Error Detection and Correction; Flow and Error Control; Sliding Window Protocol, HDLC, Multiple Access – CSMA/CD, CSMA/CA, Reservation, Polling, Token Passing, FDMA, CDMA,TDMA, Network Devices, Backbone Networks, Virtual LANs.IPv4 Structure and Address Space; Classful and Classless Addressing; Datagram, Fragmentation and Checksum; IPv6 Packet Format, Mapping Logical to Physical Address (ARP), Direct and Indirect Network Layer Delivery; Routing Algorithms, TCP, UDP and SCTP Protocols; Flow Control, Error Control and Congestion Control in TCP and SCTP**. (L-10)**

Internetworking: Switch/Hub, Bridge, Router, Gateways, concatenated virtual circuits, Tunnelling, Fragmentation**. (L-8)**

Network Security: Malwares, Cryptography and Steganography; Secret-Key Algorithms, Public-Key Algorithms, Digital Signature, Virtual Private Networks, Firewalls, Mechanisms of TLS, SSL, IPsec**. (L-10**)

Adhoc Networks: Mobile Adhoc Networks, P2P network, Wireless Transmission and Wireless LANs, semantic sensor networks. **(L-6)**

Cloud Computing and IoT: SaaS, PaaS, IaaS, Public and Private Cloud; Virtualization, Virtual Server, Cloud Storage, Database Storage, Resource Management, Service Level Agreement, Basics of IoT. **(L-10)**

References:

1.Data and Computer Communication, William Stallings. Prentice Hall of India.
2.Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI.
3.Data Communications and Networking – Behrouz A. Forouzan, Third Edition TMH.
4.An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.
5.Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson COMPUTE.

MCSCCT204: Information Security and Coding Theory

Models for Information Channel: Discrete Memoryless Channel, Binary Symmetric Channel (BSC), Burst Channel. Bit-error rates. Probability, Entropy and Shannon's measure of information. Mutual information. Channel capacity theorem. Rate and optimality of Information transmission. **(L-6)**

Variable Length Codes: Prefix Codes, Huffmann Codes, Lempel-Ziev (LZ) Codes. Optimality of these codes. Information content of these codes, Vcoder**. (L-6)**

Error Correcting and Detecting Codes: Finite fields, Hamming distance, Bounds of codes, Linear (Parity Check) codes, Parity check matrix, Generator matrix, Decoding of linear codes, Hamming codes, decoder, BCH. **(L-6)**

Cryptography: Concepts and Techniques, symmetric and asymmetric key cryptography, steganography, Symmetric key Ciphers: DES structure, DES Analysis, Security of DES, variants of DES, Block cipher modes of operation, AES structure, Analysis of AES, Key distribution Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Analysis of RSA, Diffie-Hellman Key exchange. **(L-15)**

Message Authentication and Hash Functions: Authentication requirements and functions, MAC and Hash Functions, MAC Algorithms, MD5: Secure Hash Algorithm, Whirlpool, HMAC, Digital signatures, X.509, Kerberos, IPsec, Secure Socket Layer(SSL), Transport Layer Security(TLS**). (L-15)**

Introduction to Cryptoanalysis: Linear Cryptanalysis, Differential Cryptanalysis, Cryptanalysis of **DLP. (L-6)**

Recent trends in security: IOT, Biometric, Visual Cryptography**. (L-6)**

References:

1.Principles of Information Security: Michael E. Whitman, Herbert J. Mattord, CENGAGE Learning, 4th Edition.
2.Cryptography and Network Security: William Stallings, Pearson Education,4th Edition.
3.Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 2nd Edition Atul Kahate, Cryptography and Network Security, McGraw Hill.
4.Kaufman, c., Perlman, R., and Speciner, M., Network Security, Private Communication in a public world, 2nd ed., Prentice Hall PTR., 2002.
5.Stallings, W.,.Cryptography and Network Security: Principles and Practice, 3rd ed., Prentice Hall PTR., 2003.
6.Stallings, W. Network security Essentials: Applications and standards, Prentice Hall, 2000.
7.Cryptography and Network Security; McGraw Hill; Behrouz A Forouzan Information Security Intelligence Cryptographic Principles andApp. Calabrese Thomson.
8.D. P. Nagpal, Information Security, S. Chand Company Limited.

MCSCSP205: Network Lab

Socket Programming. **(L-120)**

MCSCSP206: Database Lab

SQL. **(L-60)**
Data analysis using WEKA. **(L-60)**

Semester-III

MCSCCT301:Compiler Design

Overview of Compiler: Overview of a compiler; structure, phases and passes of compiler; problems of compiler design; application of compiler design technology; interdependencies between compiler and computer architecture. **(L-10)**

Formal Languages: Elements of formal language theory; Regular Languages; Regular grammars, regular expressions, finite state automata; conversions; state minimization. **(L-10)**

Lexical Analysis: Lexical analysis vs. parsing; tokens, lexemes; input buffering; specification and recognition of tokens; lexical analyser**.** **(L-10)**

Syntax Analysis / Parsing: Context free grammar parse trees and derivations; ambiguity; elimination of left recursion and left factoring; top-down parsing: recursive-descent parsing, predictive parsing, LL(1) parsers; bottom-up parsing: shift-reduce parsing, conflicts; LR parsing (simple, canonical, look ahead); operator precedence parser. **(L-10**)

Intermediate Code Generation: Syntax Directed Translation. Synthesized and Inherited attributes. Dependency Graph. Three Address Code Representation. Symbol table management. Syntax trees. Type checking, Control flow statements. Back-patching. **(L-10)**

Code Generation and Optimization: Directed Acyclic Graph (DAG), Runtime storage management, Peephole optimization. Compiler writing tools. **(L-10)**

References:

1. Compilers – Principles, Techniques & Tools: Aho, Sethi & Ullman, Addison Wesley.
2. Compiler Design in C: Holub, PHI.
3. Compiler Design: Dhamdhere.
4. Principles of Compiler Design: Alfred V. Aho& Jeffrey D. Ullman, Narosa.

MCSCCT302:Artificial Intelligence &Expert Systems

Introduction: Importance of Artificial Intelligence (AI**). (L-10**)

Problem solving using AI approach: Water-jug problem, travelling-salesman problem, N-Queen problem, Wampus world problem. Its representation, Organisation, Manipulation and Acquisition. First Order Predicate Logic and its use in knowledge representation, Resolution Principle. Use of Resolution in reasoning and question answering.
Introduction to PROLOG and LISP**. (L-10)**

Production Systems and Search Strategies: Production System and its variants, Search Methods, Heuristic Search Methods, AND/OR Graphs and AO* Algorithm, Searching Game Trees. **(L-10)**

Soft Computing and Uncertainty Management: Introduction of Fuzzy Logic, NN and GA; Bayesian inferencing, Dempster- Shafer theory of Beliefs. **(L-10)**

Structured Representation of Knowledge and reasoning: Semantic Networks, Frames, Scripts and Conceptual Dependency. **(L-10)**

Expert Systems (ES): Rule Based Expert System Architecture, Non-production System Architecture, Neural Network based ES, Knowledge Acquisition Methods, Explanation Methods, Case study; Mycin; Expert System Shells.Introduction to pattern Recognition, Natural Language. Processing, Planning, etc **(L-10)**

References:

1. Introduction to Artificial Intelligence& Expert System by D. W. Patterson, PHI.
2. Introduction to Artificial Intelligence by Rich & Knight.
3. Principle of Artificial Intelligence by N. J. Nilson, Narosa.

Major Elective-I (Special Paper-I)

MCSCCT303A1:Machine Learning

Basics: Introduction to machine learning - different forms of learning; Basics of probability theory, linear algebra and optimization. **(L-10)**

Classification Methods: Linear Discriminant Analysis, Large margin classification, Kernel methods, Support Vector Machines. Classification and Regression Trees, Multi-layer Perceptron's and Back propagation. **(L-10)**

Graphical Models: Bayesian Belief Networks, Markov Random Fields, Exact inference methods, approximate inference methods. **(L-10)**

Learning theory: Bias/variance trade-off. Union and Chernoff/ Hoeffdingbounds; VC dimension. Worst case (online) learning; Practical advice on how to use learning algorithms. **(L-10**)

Clustering: Partitional Clustering - k-means, k-medoids; Hierarchical Clustering - Agglomerative, Divisive, Distance measures; Density based clustering - DBScan; Spectral clustering. PCA (Principal components analysis); ICA (Independent components analysis).**(L-10)**

Reinforcement learning and control: MDPs. Bellman equations; Value iteration and policy iteration; Linear Quadratic Regulation (LQR). LQG; Q-learning. Value function approximation; Policy search. Reinforce. POMDPs. **(L-10)**

References:

1. Machine Learning - Tom Mitchell, McGraw-Hill.
2. Pattern Recognition and Machine Learning - C. M. Bishop, Springer, 2006.
3. . Pattern Classification - R. O. Duda, P. E. Hart, and D.G. Stork, John Wiley and Sons, 2001.
4. Statistical Learning Theory - Vladimir N. Vapnik, John Wiley and Sons, 1998.
5. An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods - Cristianini, N. and Shawe-Taylor, J., Cambridge University Press, 2000.
6. Introduction to Machine Learning - E. Alpaydin, Prentice Hall of India, 2006.

MCSCCP304:Compiler Design and Artificial Intelligence Lab

Compiler Design Lab: (L-60)

Programs of the types given below:

1. Write a Lex program to count the number of Lines and Characters.
2. Write a Lex program to count the number of Lines and characters in the input file.
3. Write a Lex program find the length of longest word in the input file.
4. Write a Lex program to count number of words.
5. Write a Lex specification program that generates a C program which takes a string "abcd" and prints the following output:
   abcd
   abc
   ab
   a

Artificial Intelligence Lab: (L-60)

Programs of the types given below:

1. A* algorithm, Mini Max Algorithm
2. Bag of word creation, Vectorization
3. Spam detection using NLP and Machine Learning.

MCSOET305:Open Elective(Not in Offer)

The students will have to choose a subject on offer other than the home department which will be allotted finally as per preference availability.

MCSOPP306: Outreach Programme

Outreach Programme is one kind of extension activity towards the society which will be helpful for a student in skill development. The students will visit to the nearby villages to reach to the common people of the society. The students will prepare some useful topics for presentation to the people of different age groups in the villages. The topics will be selected according to the relevance in daily life. The students will be guided by the teachers of the department. The students will try to motivate the common people about the value of education and help them to make understand the usefulness of Computer Science in daily life.

Semester-IV

MCSCCT401: Data Analytics

Fundamentals of Analytics: Need for Analytics, Advanced Data Modelling (subject oriented data), Data Warehousing, Online Analytical Processing (OLAP), Business Process Modelling. **(L-12)**

Statistical Methods and Modelling Techniques: Statistical Modelling for Computer Science, Regression, Clustering, Principal Component Analysis, Decision Theory, Foundation of Time Series Analysis, R Programming Language. **(L-20)**

Data Mining: Introduction to Data Mining Algorithms and Techniques, Supervised and Unsupervised Mining Techniques, Association Rule Mining, Classification, Clustering, Web Data Mining, Text Mining. **(L-15)**

Big Data Technology: Introduction to Big Data, Big Data Platforms, Big Data Modelling, Data Materialization and Visualization, Introduction to Big Data Analytics. **(L-13)**

References:

1. Data Science and Analytics by V K Jain, Khanna Publishing.

2. Behavioural Research Data Analysis with R by Yuelin Li, Jonathan Baron, Springer.

3. Python Data Science Handbook by Jake Vanderplas, O'Reilly.

4. Data Analytics by Anil Mahrswari, Tata Macgraw Hill.

5. Big Data: Principles and Paradigms by Rajkumar Buyya (Editor), Rodrigo N. Calheiros (Editor), Amir Vahid Dastjerdi (Editor).

MCSCCT402: Advanced Software Design

Introduction: Why object orientation, History and development of Object-Oriented Programming language, concepts of object-oriented programming language. Object, class, message passing, encapsulation, polymorphism, aggregation, threading, difference between OOP and other conventional programming-advantages and disadvantages.

Object Oriented Process Model: Fountain Model, Iterative Water Model, RUP Model, Component Based model. **(L-8)**

Object Oriented Software Analysis & Design:Object oriented analysis: Use case diagram; Major and minor elements, Object, Class. Booch, Raumbagh, Codd Yordon, Jakobson Methods.

Object oriented design: Relationships among objects, aggregation, links, relationships among classes association, aggregation, using, instantiation, meta-class, grouping constructs. **(L-6)**

Unified Modelling Language: Fundamentals of Object-Oriented design in UML: Well-formed Rules and semantic guide of UML, Structural models – Use Case Description, Class Diagram, Object diagram, Role Concepts, interaction diagram: collaboration diagram, sequence diagram,

UML Dynamic modelling concepts: state chart diagram, activity diagram, implementation diagram, UML extensibility- model constraints and comments, Note, Stereotype.

Analysis & Design of OOSE using UML: Analysis modelling using UML, Design modelling using UML, Tools support (Introduction to Rational Rose). **(L-12)**

Architecture and Quality Evaluation: Object Oriented System Architecture; Model Driven Architecture, Domain Specific Modelling notation, Model integrated Computing for OOSE.

Quality Evaluation of OOS: CK metrics and methods, Lee Metrics, Quality analysis; Validation and Verification of Object-Oriented Software**. (L-10)**

Service Oriented System: Introducing SOA: Fundamental SOA-Common characteristics of contemporary SOA- Common misperceptions about SOA- Common tangible benefits of SOA-Common pitfalls of adopting SOA The Evolution of SOA.

Web Services and Primitive SOA: The Web services framework- Services (as Web services)-Service descriptions (with WSDL)-Messaging (with SOAP), SOA Standards – OASIS Reference Model, S3, Enterprise Service Bus.

SOA and Service-Orientation: Principles of Service-Orientation, Service orientation vs object-orientation, Native Web service support for service-orientation principles. - Service Layers –Service orientation and contemporary SOA- Service layer, abstraction-application service layer-Business service layer- Orchestration service layer; Agnostic services- Service layer configuration scenarios. **(L-12)**

Agile Software Design & Development: Agile Software Development: Basics and Fundamentals of Agile Process Methods, Values of Agile, Principles of Agile, stakeholders, Challenges

Lean Approach: Waste Management, Kaizen and Kanban, add process and products add value. Roles related to the lifecycle, differences between Agile and traditional plans, differences between Agile plans at different lifecycle phases. Testing plan links between testing, roles and key techniques, principles, understand as a means of assessing the initial status of a project/ How Agile helps to build quality

Agile Requirements: User Stories, Backlog Management. Agile Architecture: Feature Driven Development. Agile Risk Management: Risk and Quality Assurance, Agile Tools

Agile Testing: Agile Testing Techniques, Test-Driven Development, User Acceptance Test

Agile Product Management: Communication, Planning, Estimation Managing the Agile Approach Monitoring progress, Targeting and motivating the team, managing business

involvement, Escalating issue. Quality, Risk, Metrics and Measurements, Managing the Agile Approach Monitoring progress, Targeting and motivating the team, managing business involvement and Escalating issue

Agile and Scrum Principles: Agile Manifesto, Twelve Practices of XP, Scrum Practices, Applying Scrum. Need of scrum, working of scrum, advanced Scrum Applications, Scrum and the Organization, scrum values. **(L-12)**

References:

1. Software Engineering a Practitioner's Approach - R S Pressman, McGraw Hill.

2. Rambaugh, James Michael, Blaha - "Object Oriented Modelling and Design" - Prentice Hall India/ Pearson Education

3. UML Standards, V 2.5, OMG, 01-03-2015

4. Robert C. Martin, Agile Software Development, Principles, Patterns, and Practices; Alan Apt Series (2011)

5. Thomas Erl ," Service-Oriented Architecture: Concepts, Technology & Design", Pearson Education Pte Ltd

Major Elective-II (Special Paper-II)

MCSMET403A2: Deep Learning

Module I:

Brief Review of Machine Learning: Types of machine learning; Process of machine learning, evaluating different ML models; Concept of loss function; Challenges faced by ML.

Basics of Neural Network: Understanding biological neuron and artificial neuron; Types of activation functions; Architectures of neural network; Learning process in ANN.

Training Deep Neural Network: Backpropagation and mathematics behind it; Deep L-layer network; Computation graph; Weight initialization in neural network; Batch, mini-batch and stochastic gradient descent; Optimization algorithms; Regularization; Normalization.

Necessary Maths for Deep Learning: Linear Algebra; Statistics; Calculus. **(L-16)**

Module II:

Computer Vision using CNN: Basic concept of computer vision; Challenges faced by traditional ANN to deal with image data; Convolutional neural network concepts – kernel,

stride, padding, pooling; Building a CNN; Popular CNN architectures – LeNet, AlexNet, GoogLeNet, ResNet, Inception network, UNET; Object detection – bounding box, YOLO, landmark detection, Transfer learning. **(L-14)**

Module III:

Representation learning: Autoencoder; Word2Vec; Self Organizing Map (SOM)

Sequence Based Models: Introduction to sequence data; Recurrent neural network; Vanishing Gradient Problem and RNN; Long Short-term Memory (LSTM); Gated Recurrent Units (GRU); Bi-directional Models; Language modelling and Sequence models

Other deep learning architectures: Generative Adversarial Network (GAN); Attention; Transformer **(L-15)**

Module IV:

Deep learning frameworks: Understanding the ecosystem of deep learning frameworks; Refresher of Python; Tensorflow 2.0 programming; PyTorch programming.

Deep Learning Case Studies (**L-15)**

References:

1. Hands-on machine learning with scikit-learn Keras and TensorFlow by AurelionGeron, published by O' Reilley.
2. Deep Learning from Scratch: Building with Python from First Principles Book by Seth Weidman.
3. Deep Learning Made Easy with R: A Gentle Itroduction for Data Science by N.D.Lewis.
4. Fundamentals of Deep Learning by Nikhil Buduma, O'REILLY.
5. Deep Learning: A Practitioner's Approach by Adam Gibson, Josh Patterson, O'REILLY.

Major Elective-III (Special Paper-III)

MCSMET404A5: Natural Language Processing

Module I:

Introduction: Natural Language Processing - Problems and perspectives

Information Extraction and Named Entity Recognition, Text processing, Corpora and their construction.

Regular Expressions and Tokenization: Regular Expression, Finite State Automata, Grammars for natural language, Word Tokenization, Normalization, Sentence Segmentation, Named Entity Recognition, Multi Word Extraction, Minimum Edit Distance, Computational Morphology, Morphological operations. **(L-15)**

Module II:

Language Modelling: Introduction to N-grams, Chain Rule, and Part of Speech Tagging – Rule based and Machine Learning based approaches

Computational Lexical Semantics: Introduction to Lexical Semantics – Homonymy, Polysemy, Synonymy, Thesaurus – WordNet, Computational Lexical Semantics – Thesaurus based and Distributional Word Similarity. **(L-15)**

Module III:

Text Classification: Text Classification, Naïve Bayes' Text Classification, Evaluation, Sentiment Analysis – Opinion Mining and Emotion Analysis, Resources and Techniques.

Information Retrieval: Boolean Retrieval, Term-document incidence, The Inverted Index, Query Optimization, Phrase Queries, Ranked Retrieval, Term Frequency, Inverse Document Frequency based ranking. **(L-17)**

Module IV:

Applications: Sentiment Analysis, Topic modelling, Summarization, Recommendation System, and Chatbot. **(L-13)**

References:

1. Allen, James, Natural Language Understanding, Second Edition, Benjamin/Cumming, 1995. Charniack, Eugene, Statistical Language Learning, MIT Press, 1993.
2. Jurafsky, Dan and Martin, James, Speech and Language Processing, Second Edition, Prentice Hall, 2008.
3. Manning, Christopher and Heinrich, Schutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
4. Radford, Andrew et. al., Linguistics, An Introduction, Cambridge University Press, 1999.

Foundation of Statistical Natural Language Processing, Manning and Schutze, MIT.

5. Natural Language Processing by Steven Bird, Ewan Klein, and Edward Loper, O'REILLIY.

MCSACT405: Add on Course

(Machine Learning, Python Programming and R Programming)

Machine Learning: (L-15)

Introduction, Basic Definitions, Types of Learning, Classification and Regression, Association Rule Mining, Clustering.

Python programming: (L-22)

Unit-I:

Basic Datatypes, Functions: definitions and use, arguments, block structure, recursion.

Unit-II:

Import, Control Structure, Boolean expressions, Iterations.

Unit-III:

Strings, Tuples, Lists, Dictionaries.

Unit-IV:

Files and Text Processing.

Unit V:

Class and Methods.

R programming: (L-23)

R Basic Fundamentals: Basic Syntax, Datatypes, Variables, Operators, Decision Making, Loops, Functions, Strings, Vectors, Lists, Matrices, Arrays, Factors, Data Frames, Decision Making, Loops, Functions, Strings, Packages.

R Data Interfaces: R CSV Files, R Excel Files, R SPSS Files, R STATA Files, R SQL.

R Graphs: Different types of R Charts and R Graphs.

R Data Management: Repeats, sorting, ordering, lists, vectors, factors, data frames.

R Data Handling: Importing Data Files, File Handling.

R Statistical Functions: Functions, basic statistical functions.

References:

1. Data Science and Analytics by V K Jain, Khanna Publishing.
2. Data Analytics by Anil Maheswari, Tata Macgraw Hill.
3. Behavioural Research Data Analysis with R by Yuelin Li, Jonathan Baron, Springer.
4. Machine Learning using Python by Manaranjan Pradhan and U Dinesh, Willey.
5. R for Data Science by Hadley Wickham and Garrett Grolem

MCSMEP406: Project and Seminar

Objectives: The project and seminar paperisa compulsory paper to all students. Some advanced topics related to special papers, or any advance topic or review work of research papers will be chosen by the students after discussion with their respective supervisor. The in-depth study of the selected topic is needed, and some advancement can be proposed in the concerned topic.

The project work and/or group project work will be performed on some advanced topics related to special papers or any advance topic or review work of research papers. The marks distribution of project work is as follows: 20 marks are allotted for written submission; 20 marks are for seminar presentation and 10 marks are allotted for viva-voce examination.

Outcomes: The typing skill of the students will be developed so as to prepare the dissertation paper. To develop the communication skill and presentation skill are also the aim of this paper. There is a scope to publish the dissertation papers in journals also.