# Academic Handbook
## *for*

# Master of Technology Programme

## *in*

# Computer Science and Engineering

Department of Computer Science and Engineering

National Institute of Technology Goa

Farmagudi, Ponda, Goa - 403 401

# Semester-wise Credit Distribution

| Semester | Total Credits |
|---|---|
| I | 18 |
| II | 18 |
| III | 14 |
| IV | 14 |
| **Total Credits** | **64** |

# Summary of Course Contents (First year)

## First Semester

| Sl. No | Sub. Code | Subjects | L-T- P | Credits |
|---|---|---|---|---|
| 1 | CS600 | Advanced Algorithms & Analysis (AAA) | 3-0-0 | 3 |
| 2 | CS601 | Advanced Computer Networks (ACN) | 3-0-0 | 3 |
| 3 | CS602 | Mathematical Foundations of Computer Science (MFCS) | 3-0-0 | 3 |
| 4 | CS8** | Elective-I | 3-0-0 | 3 |
| 5 | CS603 | Advanced Algorithms & Analysis Laboratory | 0-0-3 | 2 |
| 6 | CS604 | Advanced Computer Networks Laboratory | 0-0-3 | 2 |
| 7 | CS605 | Seminar | 0-0-3 | 2 |
| | | **Total Credits** | | **18** |

## Second Semester

| Sl. No | Sub. Code | Subjects | L-T- P | Credits |
|---|---|---|---|---|
| 1 | CS650 | Advanced Database Systems (ADBS) | 3-0-0 | 3 |
| 2 | CS651 | Advanced Computer Architecture (ACA) | 3-0-0 | 3 |
| 3 | CS652 | Object Oriented Software Engineering (OOSE) | 3-0-0 | 3 |
| 4 | CS8** | Elective-II | 3-0-0 | 3 |
| 5 | CS653 | Advanced Database Systems Laboratory | 0-0-3 | 2 |
| 6 | CS654 | Object Oriented Software Engineering Laboratory | 0-0-3 | 2 |
| 7 | CS655 | VIVA | - | 2 |
| 8 | HU650 | Communication Skills and Technical Writing (Audit Course) | 1-0-2 | - |
| | | **Total Credits** | | **18** |

# Summary of Course Contents (Second year)

| Third Semester | | | | |
|---|---|---|---|---|
| **Sl. No** | **Sub. Code** | **Subjects** | **L-T- P** | **Credits** |
| 1 | CS700 | Major Project-I | 0-0-12 | 8 |
| 2 | CS8** | Elective-III | 3-0-0 | 3 |
| 3 | CS8** | Elective-IV | 3-0-0 | 3 |
| | | **Total Credits** | | **14** |

| Fourth Semester | | | | |
|---|---|---|---|---|
| **Sl. No** | **Sub. Code** | **Subjects** | **L-T- P** | **Credits** |
| 1 | CS750 | Major Project-II | 0-0-21 | 14 |
| | | **Total Credits** | | **14** |

**NOTE:** The applicant must have a sufficient background in computer science and engineering to complete the degree requirements with reasonable performance. As the students with background other than computer science and engineering have been allowed to apply for M. Tech programme in computer science and engineering, they may not be allowed to get the admission into the programme, if they are not exposed to the prerequisites such as below.

| Prerequisites for the Admission into the Programme | |
|---|---|
| 1 | Data Structures |
| 2 | Design and Analysis of Algorithms |
| 3 | Computer Organization and Architecture |
| 4 | Discrete Mathematics |
| 5 | Computer Networks |
| 6 | Database Management Systems |

# List of Electives

| Program Specific Electives | | | | |
|---|---|---|---|---|
| **SI. No.** | **Course Code** | **Course Name** | **Total Credits** (L-T-P) | **Credits** |
| 1 | CS800 | Foundations of Cryptography | (3-0-0) | 3 |
| 2 | CS801 | Wireless Sensor Networks | (3-0-0) | 3 |
| 3 | CS802 | Advanced Compiler Design | (3-0-0) | 3 |
| 4 | CS803 | Distributed Computing Systems | (3-0-0) | 3 |
| 5 | CS804 | Design of Secure Protocols | (3-0-0) | 3 |
| 6 | CS805 | Mobile Computing | (3-0-0) | 3 |
| 7 | CS806 | Machine Learning | (3-0-0) | 3 |
| 8 | CS807 | Health Informatics | (3-0-0) | 3 |
| 9 | CS808 | Soft Computing | (3-0-0) | 3 |
| 10 | CS809 | Service Oriented Architecture & Cloud Computing | (3-0-0) | 3 |
| 11 | CS810 | Big Data Analytics | (3-0-0) | 3 |
| 12 | CS811 | Pattern Recognition | (3-0-0) | 3 |
| 13 | CS812 | Artificial Neural Networks | (3-0-0) | 3 |
| 14 | CS813 | Computer Vision | (3-0-0) | 3 |
| 15 | CS814 | Game Theory | (3-0-0) | 3 |
| 16 | CS815 | Data Warehousing & Data Mining | (3-0-0) | 3 |
| 17 | CS816 | E-Commerce | (3-0-0) | 3 |
| 18 | CS817 | Advanced Operating Systems | (3-0-0) | 3 |
| 19 | CS818 | Security and Privacy | (3-0-0) | 3 |
| 20 | CS819 | Bioinformatics Algorithms | (3-0-0) | 3 |
| 21 | CS820 | Graph Theory | (3-0-0) | 3 |
| 22 | CS821 | Probability and Statistics | (3-0-0) | 3 |
| 23 | CS822 | Program Analysis and Verification | (3-0-0) | 3 |
| 24 | CS823 | Linear Algebra | (3-0-0) | 3 |
| 25 | CS824 | Number Theory | (3-0-0) | 3 |
| 26 | CS825 | Complexity Theory | (3-0-0) | 3 |
| 27 | CS826 | Human Computer Interface | (3-0-0) | 3 |

# 1-Credit Module Courses

| SI. No. | Course Code | Course Name | Total Credits (L-T-P) | Credits |
|---|---|---|---|---|
| 1 | CS827 | Special Module in Computational Geometry | (3-0-0) | 1 |
| 2 | CS828 | Special Module in Parallel Computation | (3-0-0) | 1 |
| 3 | CS829 | Special Module in Hardware Systems | (3-0-0) | 1 |
| 4 | CS830 | Special Module in Theoretical Computer Science | (3-0-0) | 1 |
| 5 | CS831 | Special Module in Artificial Intelligence | (3-0-0) | 1 |
| 6 | CS832 | Special Module in High Speed Networks | (3-0-0) | 1 |
| 7 | CS833 | Special Module in Concurrency | (3-0-0) | 1 |
| 8 | CS834 | Special Module in NLP | (3-0-0) | 1 |
| 9 | CS835 | Special Module in Numerical Methods | (3-0-0) | 1 |
| 10 | CS836 | Special Module in CSE* | (3-0-0) | 1 |

These courses will usually cover topics that are not generally covered in the regular courses. Interested students can register for these courses for credits, provided, the above semester-wise credit structure is followed. They are evaluated like any other courses and credits earned count towards degree requirements. The syllabi of these courses are not specified. It will be decided by the courses instructor from time to time. These courses can be given anytime in the semester. They are specially designed to take advantage of short time eminent visitors from Industry/Academics.

* The 1-credit module course CS836 will cover topics of current interest in computer science and engineering.

# Audit Course*

| Sl. No | Sub. Code | Subjects | L-T- P | Credits |
|---|---|---|---|---|
| 1 | HU650 | Communication Skills and Technical Writing | 1-0-2 | - |

**(* No credits)**

# Proposed Course Contents

<table>
<tr><td>Subject Code<br>CS600</td><td colspan="2">Advanced Algorithms & Analysis (AAA)</td><td>Credits: 3 (3-0-0)<br>Total hours: 45</td></tr>
<tr><td>Course Objectives</td><td colspan="3">To study paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.</td></tr>
<tr><td colspan="3">Module 1</td><td>5 Hours</td></tr>
<tr><td colspan="4">Formal models of computation, time and space complexity, Proof of lower. Algorithm design techniques: Greedy algorithms, divide-and-conquer algorithms, dynamic programming, Branch-and-bound, amortization, optimal algorithms.</td></tr>
<tr><td colspan="3">Module 2</td><td>15 Hours</td></tr>
<tr><td colspan="4">Algorithms on arrays: Selection and median-finding, counting, radix and bucket sorts, string matching (Rabin-Karp and Knuth-Morris-Pratt algorithms) etc.,<br>Geometric algorithms : Convex hulls, sweep paradigm, Voronoi diagrams. etc.,<br>Algorithms on graphs : Traversal, topological sort, minimum spanning trees, shortest path, network flow -preflow-push algorithms, max flow algorithm etc.,<br>Arithmetic algorithms : GCD, modular arithmetic, primality testing etc., Numerical algorithms, Internet algorithms.</td></tr>
<tr><td colspan="3">Module 3</td><td>10 Hours</td></tr>
<tr><td colspan="4">NP-Completeness: Polynomial time, Verification, NP-Completeness and reducibility, NP-Completeness proofs, NP-Complete problems.</td></tr>
<tr><td colspan="3">Module 4</td><td>10 Hours</td></tr>
<tr><td colspan="4">Randomized algorithms: Monte Carlo and Las Vegas algorithm. Randomized algorithms for the problems in various domains viz., Graph algorithms, Geometric algorithms parallel and distributing algorithms, online algorithms, Number theory and algebra., etc.,</td></tr>
<tr><td colspan="3">Module 5</td><td>5 Hours</td></tr>
<tr><td colspan="4">Approximation Algorithms: PTAS and FPTAS algorithms, Combinatorial algorithms- Setcover, cut , TSP etc. Exact exponential algorithms:</td></tr>
<tr><td>Reference Books</td><td colspan="3">1. T. Cormen, Charles E. Leiserson and Ronald D River, Introduction to Algorithms, PHI, 3rd edition, 2009.<br>2. Aho, Hopcroft and Ullman The design and analysis of Computer Algorithms, Addison Weseley, 1st edition, 1974.<br>3. M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, 1st edition, 1979.<br>4. Rajeev Motwani and Prabhakar Raghavan, Randomized Algorithms, Cambridge University, 1st edition, 1995.<br>5. Vijay V Vazirani, Approximation Algorithms, Springer, 2002.</td></tr>
</table>

| | | |
|---|---|---|
| **Subject Code CS601** | **Advanced Computer Networks (ACN)** | **Credits: 3 (3-0-0)**<br>**Total hours: 45** |
| **Course Objectives** | To understand the theoretical and the practical aspects of the advanced networking principles including the distributed computing. The course involves the future networking principles also. | |
| **Module 1** | | **10 Hours** |
| Review of Basic Network Architectures: OSI reference model, TCP/IP reference model, ATM reference model; Applications(WWW, Audio/Video Streaming, Video conference, Networked Games, Client/Server); Traffic Characterization (CBR, VBR); Switching Paradigms; Multiplexing; Error Control; Flow Control, FTH, DTH, PON, ISDN, DSL, CATV, SONET, Optical Networks. | | |
| **Module 2** | | **8 Hours** |
| Local Area Network Technologies: Fast Ethernet, Gigabit Ethernet, IEEE 802.11 WLAN, Bluetooth, Connecting LANs, VLANS. | | |
| **Module 3** | | **10 Hours** |
| Internetworking: Interdomain Routing, BGP, IPv6, Multicast Routing Protocols, Multi Protocol Label Switching, Virtual Private Networks, High speed transport protocols, Quality of Service Mechanisms, Improving QoS in Internet, DiffServ and IntServ Architectures, RSVP. | | |
| **Module 4** | | **12 Hours** |
| Distributed Systems: Naming, DNS, DDNS, Paradigms for Communication in Internet, Caching, Issues of Scaling in Internet and Distributed Systems, Caching Techniques forWeb, Protocols to Support Streaming Media, Multimedia Transport Protocols, Content Delivery Networks, Overlay and P2P Networks. | | |
| **Module 5** | | **5 Hours** |
| Applications and Other Networking Technologies: RTP, RTSP, SIP, VoIP, Security Systems, SSH, PGP, TLS, IPSEC, DDoS Attack, Mitigation in Internet, Security in MPLS; Introduction to Cellular, Satellite and Ad hoc Networks. | | |
| **Reference Books** | 1. Behrouz A. Forouzan, *Data Communications and Networking*, 5th edition, Tata McGraw Hill, 2013.<br>2. Larry L. Peterson and Bruce S. Davie, *Computer Networks: A Systems Approach*, 4th edition, Morgan Kaufmann, 2007.<br>3. J. Walrand and P. Varaiya, *High Performance Communication Networks*, 2nd edition, Morgan Kauffman, 2000<br>4. Markus Hoffmann and Leland R. Beaumont, *Content Networking: Architecture, Protocols, and Practice*, Morgan Kauffman, 2005. | |

| | | |
|---|---|---|
| **Subject Code**<br>**CS602** | **Mathematical Foundations of Computer Science (MFCS)** | **Credits: 3 (3-0-0)**<br>**Total hours: 45** |
| **Course Objectives** | This course introduces the mathematical foundations for computer science, viz., Mathematical logic, Combinatorics, Boolean and linear algebra and Automata theory. | |
| **Module 1** | | **10 Hours** |
| Language of Math– Logic, Proof techniques, (infinite) sets, countable and uncountable sets, Functions, Relations, Cantor's diagonalization, Applications to undecidability, Induction, Recursion, Basic number theory: Divisibility, congruences, quadratic residues. | | |
| **Module2** | | **6 Hours** |
| Combinatorics– General Counting methods, Recurrence relations, Generating Functions, Principle of Inclusion-Exclusion, Posets and Lattices - Permutations, Groups and algebraic structures. | | |
| **Module 3** | | **09 Hours** |
| Automata, Grammars and Languages: Regular languages and finite automata, Context-free languages and pushdown automata, Turing machines, Some other computing models and formalisms, their equivalence with Turing machines, Undecidability. | | |
| **Module 4** | | **10 Hours** |
| Probability– Sample space, Distributions, Random Variables, Expectation, Tail Inequalities - Chernoff Bound, Chebyshev inequality, Functions of random variables, Applications. | | |
| **Module 5** | | **10 Hours** |
| Linear Algebra– Fields, Vector Spaces, Basis, Matrices and Linear Transformations, Eigen values, Orthogonality, Vector and Matrix Norms - Applications to optimization problems and graph theory. | | |
| **Reference Books** | 1. W. Feller, *An Introduction to Probability Theory and Its Applications*, Wiley; vol. 1 & 2, 1971.<br>2. Jean Gallier, *Discrete mathematics*, Springer, 2011.<br>3. John Hopcroft, Rajeev Motowani and Jeffrey Ullman, *Automata Theory, Languages, and Computation*, 3rd edition, 1974.<br>4. Gilbert Strang, *Introduction to Linear Algebra*, 4th Edition, Wellesley-Cambridge Press, Wellesley, MA, 2009. | |

| Subject Code<br>CS603 | **Advanced Algorithms & Analysis Laboratory** | Credits: 2 (0-0-3)<br>Total hours: 42 |
|---|---|---|
| **Course Objectives** | To have hands on session with the algorithms. | |

**Experiments**

Experiments include the implementations of the algorithms related to various computational problems in various domains viz., Graph algorithms, Geometric algorithms, cryptographic algorithms, and numerical algorithms etc., using different design paradigms.

- Divide and conquer algorithms.
- Greedy algorithms
- Dynamic programming algorithms
- Branch and bound algorithms

Implementation of the randomized algorithms for various computational problems and comparison with their best deterministic counterparts.

**Reference Books**

1. T. H. Cormen, C. L. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd edition, MIT Press, 2009.
2. Harry R. Lewis and Larry Denenberg, *Data Structures and Their Algorithms*, Harper Collins, 1st edition, 1991.
3. Michael T. Goodrich and Roberto Tamassia, *Algorithm Design: Foundations, Analysis, and Internet Examples*, 2nd edition, John Wiley, 2008.
4. M. H. Alsuwaiyel, *Algorithm Design Techniques and Analysis*, vol. 7, World Scientific, 1999.
5. Sara Baase and Allen Van Gelder, *Computer Algorithms: Introduction to Design and Analysis*, Addison-Wesley, 2000.

| **Subject Code CS604** | **Advanced Computer Networks Laboratory** | **Credits: 2 (0-0-3)**<br>**Total hours: 42** |
|---|---|---|
| **Course Objectives** | To provide hands on in the topics studied in advanced computer networks course | |
| This laboratory focuses on developing applications inter process communication tools such as pipes, FIFOs, message queues and sockets. Broadly applications will be of the following nature:<br>1. Developing basic network client server programs to exchange data, stream audio and video<br>2. To develop a chat application<br>3. To develop a networked multi-party game<br>4. Simulation of the routing algorithms<br>5. Exercises to explore transport protocols<br>6. Simulation of the distributed systems<br>7. Running clock synchronization algorithms | | |
| **Reference Books** | 1. Larry L. Peterson and Bruce S. Davie, *Computer Networks: A Systems Approach*, 4th edition, Morgan Kaufmann, 2007.<br>2. W. Richard Stevens, Bill Fenner and Andrew M. Rudoff, *UNIX Network Programming*, 3rd edition, Addison Wesley, 2003.<br>3. Elliotte Rusty Harold, *Java Network Programming*, 3rd edition, O'Reilly, 2004. | |

| **Subject Code**<br><br>**CS605** | **Seminar** | **Credits: 2 (0-0-3)** |
|---|---|---|
| **Course Objectives** | Students will have to choose a topic in CSE's current trends or industry practices, prepare a write up, and present it along with a suitable demonstration. | |

| | | |
|---|---|---|
| **Subject Code CS650** | **Advanced Database Systems (ADBS)** | **Credits: 3 (3-0-0)** **Total hours: 45** |
| **Course Objectives** | To develop an appreciation of emerging database trends as they apply to semi-structured data, the internet, and object-oriented databases. To explain the process of DB Query processing and evaluation. To understand and evaluate the aspects of database security. | |
| **Module 1** | | **8 Hours** |
| Distributed database concepts, overview of client-server architecture and its relationship to distributed databases, concurrency control heterogeneity issues, persistent programming languages, object identity and its implementation, clustering, indexing, client server object bases, cache coherence. | | |
| **Module 2** | | **10 Hours** |
| Parallel databases: Parallel architectures, performance measures, shared nothing/shared disk/shared memory based architectures, data partitioning, intra-operator parallelism, pipelining, scheduling, load balancing, query processing- index based, query optimization: cost estimation, query optimization: algorithms, online query processing and optimization, XML, DTD, XPath, XML indexing, adaptive query processing. | | |
| **Module 3** | | **10 Hours** |
| Advanced transaction models: Save points, sagas, nested transactions, multi-level transactions, Recovery, multilevel recovery, shared disk systems, distributed systems 2PC, 3PC, replication and hot spares, data storage, security and privacy- multidimensional k- anonymity, data stream management. | | |
| **Module 4** | | **8 Hours** |
| Models of spatial data: Conceptual data models for spatial databases (e.g. pictogram enhanced ERDs), logical data models for spatial databases: raster model (map algebra), vector model, spatial query languages, need for spatial operators and relations, SQL3 and ADT. spatial operators, OGIS queries. | | |
| **Module 5** | | **9 Hours** |
| Access Control-Models, Policy. Trust management and Negotiations, Secure data outsourcing, Security in Advanced Database systems, Security in Data Warehouses and OLAP systems, Spatial database security, Security for workflow systems, Database watermarking. | | |
| **Reference Books** | 1. AviSilberschatz, Henry Korth, and S. Sudarshan, *Database system concepts*, 5th edition, McGraw Hill, 2005.<br>2. R. Elmasri and S. Navathe, *Fundamentals of database systems*, Benjamin - Cummings,5th edition, 2007.<br>3. Ceri S and Pelagatti G, *Distributed databases principles and systems*, 2nd edition, Mc-Graw Hill, 1999.<br>4. S. Castino, M. Fugini, G. Martella and P. Samarati (eds), *Database Security*, Addison Wesley, 1994.<br>5. Michael Gertz, Sushil Jajodia, *Handbook of Database Security: Applications and Trends*, Springer, 2008. | |

| Subject Code CS651 | Advanced Computer Architecture (ACA) | Credits: 3 (3-0-0) Total hours: 45 |
|---|---|---|
| **Course Objectives** | To understand the design principles of the modern computing systems | |

**Module 1** — **10 Hours**

Principles of computer organization: Data representation, data path design- pipelined arithmetic unit design, representation of instructions- instruction set architectures (RISC and CISC), instruction format, design of the control unit; memory hierarchy design-basic memory cell, memory chip, memory unit, cache memory unit design with mapping methods and multi-level cache design, design of memory management unit; I/O methods

Pipelined processor design: overlapped execution of instructions, pipeline hazards, pipeline idealism,

**Module 2** — **13 Hours**

Superscalar processor: Parallel pipelines, instruction level parallelism, out of order execution of instructions, semantic constraints: register data flow techniques, memory data flow techniques, control flow techniques, dynamic techniques.

**Module 3** — **11 Hours**

High performance computing architectures: Parallel computer models and program parallelism, Classification of machines, SISD, SIMD and MIMD, Conditions of parallelism, data and resource dependencies, hardware and software parallelism, program partitioning and scheduling, grain size latency, program flow mechanism, control flow versus data flow, data flow architecture, demand driven mechanisms, comparison of flow mechanisms.

**Module 4** — **11 Hours**

Advanced processor architectures: Multithreaded processors, multi-core processors, multi-processor systems, cache-coherence protocols, directory based protocols. Storage systems: strage area networks, RAID architecture, Graphics processing units.

**Reference Books**


1. John Paul Shen and Mikko H. Lipasti, *Modern processor design - Fundamentals of superscalar processors*, Tata McGraw Hill, 2005.
2. V. Rajaraman and C. Sivarama murthy, *Parallel Computer: Architecture and Programming*, PHI, 2000.
3. K. Hwang and F.A. Briggs, *Computer Architecture and Parallel Processing*, McGraw Hill, 1984.
4. John L. Hennesy and David A. Patterson, *Computer Architecture- A quantitative approach*, 4th edition, Elsevier, 2007
5. Dezso Sima, Terence Fountain and Peter Kacsuk, *Advanced Computer Architectures*: *A design space approach*, Addison Wesley, 1997.
6. John P. Hayes, *Computer Architecture and Organization*, 3rd edition, McGraw Hill, 1998.

| Subject Code<br>CS652 | **Object Oriented Software Engineering (OOSE)** | **Credits: 3 (3-0-0)**<br>**Total hours: 45** |
|---|---|---|
| **Course Objectives:** | This course introduces Object-oriented software engineering (OOSE) - which is a popular technical approach to analyzing, designing an application, system, or business by applying the object-oriented paradigm and visual modeling. | |
| **Module 1** | | **11 Hours** |
| Introduction to software engineering - software engineering concepts, software engineering development activities, managing software development, project organization and communication; Introduction to UML - UML notations – package diagrams, component diagrams, deployment diagrams, use-case diagrams, activity diagrams, class diagrams, sequence diagrams, interaction overview diagrams, composite structure diagrams, state machine diagrams, timing diagrams, object diagrams, communication diagrams. | | |
| **Module 2** | | **12 Hours** |
| Requirements elicitation - functional and nonfunctional requirements, completeness, consistency, clarity and correctness, realism, verifiability and traceability; requirements elicitation activities – identifying actors, scenarios, use-cases; maintaining traceability and documentation. Analysis modeling – analysis object models and dynamic models, entity, boundary and control objects, generalization and specialization; analysis activities – from use cases to objects, managing and documenting analysis. | | |
| **Module 3** | | **11 Hours** |
| System design concepts – subsystem and classes, services and subsystem interfaces, coupling and cohesion, layers and partitions; system design activities – from objects to subsystems; addressing design goals – mapping subsystems to processors and components, identifying and storing persistent data, providing access control, designing the global control flow, identifying services and boundary conditions; managing and documenting system design; object design – specifying interfaces. | | |
| **Module 4** | | **11 Hours** |
| Mapping models to code – model transformation, refactoring, forward engineering, reverse engineering, transformation principles; mapping activities; managing implementation; testing concepts – faults, erroneous states and failures; testing activities – component inspection, usability testing, unit testing, integration testing, system testing; managing and documenting testing. Rationale management, configuration management, project management, software lifecycle. | | |
| **Reference Books** | 1. Bernd Bruegge and Allen H. Dutoit, *Object-Oriented Software Engineering Using UML, Patterns, and Java*, 3rd edition, Pearson Education, 2009.<br>2. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen and Kelli A. Houston, *Object-Oriented Analysis and Design with Applications*, 3rd edition, Addison-Wesley.<br>3. Mike O'Docherty, *Object-Oriented Analysis and Design: using UML*, Wiley Publication, 2005.<br>4. Alan Dennis, Barbara Haley Wixom and David Tegarden, *Systems Analysis and Design with UML 2.0 - An Object-Oriented Approach*, 4th edition, Wiley, 2012. | |

<table>
<tr><td>Subject Code<br><br>CS653</td><td>Advanced Database Systems Laboratory</td><td>Credits: 2 (0-0-3)<br>Total hours: 42</td></tr>
<tr><td>Course Objectives</td><td colspan="2">To have hands on session of the Database concepts</td></tr>
<tr><td colspan="3">1. Database schema design<br><br>2. Database creation,<br><br>3. SQL programming and report generation using a commercial RDBMS like ORACLE/SYBASE/DB2/SQL-Server/INFORMIX.<br><br>4. Students are to be exposed to front end development tools, ODBC and CORBA calls from application Programs.<br><br>5. Internet based access to databases and database administration.<br><br>6. A project on distributed databases (decided by the instructor.)<br><br>7. Implementation of Role based model for a database system.<br><br>8. Database security exercises.</td></tr>
<tr><td>Reference Books</td><td colspan="2">1. AviSilberschatz, Henry Korth, and S. Sudarshan, Database system concepts, 5th edition, McGraw Hill, 2005.<br>2. Ralf HartmutGuting and Markus Schneider, Moving objects databases, Morgan Kaufman, 2005.<br>3. R. Elmasri and S. Navathe, Fundamentals of database systems, 5th edition Benjamin - Cummings, 2007.<br>4. Raghu Ramakrishnan, Database management systems, McGraw-Hill, 2000.<br>5. Ceri S and Pelagatti G, Distributed databases principles and systems, 2nd edition, Tata Mc-Graw Hill, 1999.</td></tr>
</table>

<table>
<tr><td>Subject Code<br>CS654</td><td>Object Oriented Software Engineering (OOSE) Laboratory</td><td>Credits: 2 (0-0-3)<br>Total hours: 42</td></tr>
<tr><td>Course Objectives</td><td colspan="2">The participants are expected to analyze application scenarios and design information systems using the Unified Modeling Language (UML). Furthermore, the designed systems are to be implemented using object-oriented programming language such as Java.</td></tr>
<tr><td colspan="3">Select domain of interest (e.g. e-Commerce) and identify multi-tier software application to work on (e.g. e-Ticketing). Analyze, design and develop this application using OOSE approach:<br>1. Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).<br>2. Identify use cases and develop the use case model.<br>3. Identify the business activities and develop an UML Activity diagram.<br>4. Identity the conceptual classes and develop a domain model with UML Class diagram.<br>5. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.<br>6. Draw the state chart diagram.<br>7. Identify the user interface, domain objects, and technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.<br>8. Implement the technical services layer.<br>9. Implement the domain objects layer.<br>10. Implement the user interface layer.<br>11. Draw component and deployment diagrams.<br>Suggested Software Tools: ArgoUML, Eclipse IDE, Visual Paradigm for UML, StarUML, and Rational Software Architect.</td></tr>
<tr><td>Reference Books</td><td colspan="2">1. Bernd Bruegge and Allen H. Dutoit , Object-Oriented Software Engineering Using UML, Patterns, and Java, 3rd edition, Pearson Education, India, 2009.<br>2. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen and Kelli A. Houston, Object-Oriented Analysis and Design with Applications, 3rd edition, Addison-Wesley, 2007.<br>3. Mike O'Docherty, Object-Oriented Analysis and Design: using UML, John Wiley & Sons, 2005.<br>4. Alan Dennis, Barbara Haley Wixom and David Tegarden, Systems Analysis and Design with UML 2.0 - An Object-Oriented Approach, 4th edition, Wiley, 2012.</td></tr>
</table>

| Subject Code CS655 | VIVA | Credits: 2 |
|---|---|---|
| **Course Objectives** | Students will have to attend for a viva-voce in presence of all the faculty of the department for the evaluation of the subjects studied in the first year (I and II semesters) with a suitable demonstration. | |

| **Subject Code**<br>**HU650 (Audit Course)** | **Communication Skills and Technical Writing** | **Credits: 0 (1-0-2)**<br>**Total hours: 45** |
|---|---|---|
| **Course Objectives** | This course is meant for developing Professional Communication and Technical Writing Skills among the students. The Lab hours will give emphasis on Technical Presentation and Seminar (on different emerging topics) followed by question-answer and discussion. | |
| **Module 1** | | **12 hours** |
| Introduction to Communication-Definition-Types-Classifications, Writing Exercises-Paragraph- Précis-Summary/Executive Summary/Abstract | | |
| **Module 2** | | **8 hours** |
| Technical Reports-Types-Format-Nuances to be followed | | |
| **Module 3** | | **10 hours** |
| Preparation of Technical Document-Reports-Instruction Manuals-Project Proposal (Prefatory Part- Main Part- Terminal Section) | | |
| **Module 4** | | **15 hours** |
| Presentation of Technical Report (Kinesics, Proxemics, and Professional Ethics ) | | |
| **Reference Books:** | 1. Raman and Sharma, *Communication Skills*, New Delhi: OUP, 2011.<br>2. Mandel, Steve, *Technical Presentation Skills: A Practical Guide for Better Speaking* (Revised Edition), Crisp Learning, 2000.<br>3. Wood, Millett, *The Art of Speaking*, New York: Drake Publishers, 1971.<br>4. Lencioni, Patrick, *The Five Dysfunctions of a Team*: NJ, John Wiley and Sons, 2006. | |

| | | |
|---|---|---|
| **Subject Code CS800** | **Foundations of Cryptography (FC)** | **Credits: 3 (3-0-0)**<br>**Total hours: 45** |
| **Course Objectives** | The purpose of the course is to familiarize the students to the arithmetic topics that have been at the centre of interest in applications of number theory, particularly in cryptography. It also includes familiarizing the students with cryptography, cryptographic protocols and the latest elliptic curve systems. | |
| **Module 1** | | **12 Hours** |
| Mathematical preliminaries: Number theory and algebra, Finite fields. | | |
| **Module 2** | | **6 Hours** |
| Symmetric key encryption: Stream ciphers and block ciphers. | | |
| **Module 3** | | **10 Hours** |
| Public key cryptography, Digital signatures, Attacks, Hash functions, Authentication schemes, Key exchange algorithm, Public key infrastructure. | | |
| **Module 4** | | **8 Hours** |
| Identification schemes, Interactive proofs, Commitment protocols, Zero knowledge proofs, Non-interactive proofs. | | |
| **Module 5** | | **9 Hours** |
| Secret sharing schemes, Digital cash, Electronic voting, Elliptic curve, Elliptic curve cryptosystems, Identity based encryption. | | |
| **Reference Books** | 1. Neal Koblitz, *Number theory and cryptography*, Springer, 2007.<br>2. Hans Delfs, Helmut Knebl, *Introduction to Cryptography: Principles and Applications*, Springer, 2002.<br>3. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.<br>4. Rudolf Lidl, Herald Niederreiter, *Introduction to Finite Fields and their Applications*, Cambridge University Press, 1994.<br>5. Ivan Niven, Herbert S. Zukerman, Hugh L.Montgomery, *An Introduction to the Theory of Numbers*, John Wiley, 1991. | |

| **Subject Code**<br><br>**CS801** | **Wireless Sensor Networks (WSN)** | | **Credits: 3 (0-0-3)**<br><br>**Total hours: 45** |
|---|---|---|---|
| **Course Objectives:** | A wireless sensor network (WSN) is a network of spatially distributed autonomous sensors those monitor physical or environmental conditions and cooperatively pass their data through the network to a main location. This course introduces the wireless sensor networks technology and discusses challenges in the design and management of wireless sensor networks. | | |
| **Module 1** | | | **9 hours** |
| Introduction to WSN, WSN applications - structural health monitoring, traffic control, healthcare, pipeline monitoring, precision agriculture, active volcano, underground mining, sensor node architecture and operating systems. | | | |
| **Module 2** | | | **11 hours** |
| WSN architectural framework, physical layer – source encoding, channel encoding, modulation, signal propagation, wireless MAC protocols – energy efficiency, scalability, adaptability, low latency and predictability, reliability, network layer – routing metrics, flooding and gossiping, routing protocols. | | | |
| **Module 3** | | | **11 hours** |
| Node and network management, power management – local power management, dynamic power management, time synchronization in WSN – basics and protocols, localization – ranging techniques, range-based localization, range-free localization, event-driven localization, WSN security – fundamentals and challenges, security attacks, protocols and mechanisms for security. | | | |
| **Module 4** | | | **14 hours** |
| Sensor network programming, radio basics, introduction to ZigBee – network topology, addressing basics, PAN addresses, channels, basic ZigBee chat, advanced ZigBee, introduction to Arduino, serial flow control, building WSN with Zigbee and Arduino, IEEE 802.15.4 and ZigBee security. | | | |
| **Reference Books** | 1. Ian F. Akyildiz, Mehmet Can Vuran, *Wireless Sensor Networks in Communications and Networking*, Wiley, 2011.<br>2. Robert Faludi , *Building Wireless Sensor Networks: with ZigBee, XBee, Arduino, and Processing*, O'Reilly Media, 2010.<br>3. Ibrahiem M. M. El Emary, S. Ramakrishnan, *Wireless Sensor Networks: From Theory to Applications*, CRC Press, 2013.<br>4. Waltenegus Dargie, Christian Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*, Wiley-Blackwell, 2010. | | |

| Subject Code CS802 | **Advanced Compiler Design (ACD)** | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives** | Describe the steps and algorithms used by language translators, Recognize the underlying formal models such as finite state automata, push-down automata and their connection to language definition through regular expressions and grammars, Discuss the effectiveness of optimization. To understand the advancements in compiler construction. | |
| **Module 1** | | **6 Hours** |
| Introduction to compiler design, Model of a Compilers, Translators, Interpreters, Assemblers, Languages, Computer Architecture vs Compiler Design, Lexical analyzer, Regular expressions and finite automata. | | |
| **Module2** | | **6 Hours** |
| Introduction to context free grammars, BNF notation, Syntax Analysis. | | |
| **Module 3** | | **8 Hours** |
| Parsing Techniques: Top-down parsing and Bottom-up parsing, general parsing strategies, brute force approach, recursive descent parser and algorithms, simple LL(1) grammar, bottom-up parsing-handle of a right sentential form, shift reduce parsers, operator precedence parsers, LR, SLR, Canonical LR, LALR grammar and parsers, error recover strategies for different parsing techniques. | | |
| **Module 4** | | **10 Hours** |
| Symbol table, syntax-directed translation schemes, intermediate code generation, translation schemes for programming language constructs, runtime storage allocation. Code generation, improvement and instruction selection: Issues, basic blocks and flow graphs, register allocation, DAG representation of programs, code generation from DAG, peep hole optimization, dependence analysis and redundancy elimination, specifications of machine. | | |
| **Module 5** | | **15 Hours** |
| Code optimization: source of optimizations, optimization of basic blocks, loops, global dataflow analysis, procedural and inter-procedural optimization, instruction scheduling optimization for memory hierarchy, solution to iterative dataflow equations. Compilation for high performance architecture; Portability and retargetability, Selected topics from compilers for imperative, object-oriented and mark-up languages, parallel and distributed programming and concurrency. | | |
| **Reference Books** | 1. Alfred V. Aho, Ravi Sethi & Jeffrey D. Ullman, *Compilers; Principles, Techniques & Tools*, Addison- Wesley Publication, 2001.<br>2. William A. Barrett, John D. Couch, *Compiler Construction, Theory and Practice*, Galgotia, 2000.<br>3. Steven S. Muchnik, *Advanced Compiler Design & Implementation*, Morgan Kaufmann Publishers, 1997.<br>4. Michael L. Scott, *Programming Language Pragmatics*, Morgan Kaufmann, 2009.<br>5. Randy Allen and Ken Kennedy, *Optimizing Compilers for Modern Architectures*, Morgan Kaufmann, 2001. | |

| Subject Code<br><br>**CS803** | **Distributed Computing Systems (DCS)** | **Credits: 3 (3-0-0)**<br><br>**Total hours: 45** |
|---|---|---|
| **Course Objectives** | This course covers abstractions and implementation techniques for the design of distributed systems. It focuses on server design, network programming, naming, storage systems, security, and fault tolerance. | |
| **Module 1** | | **10 Hours** |
| Introduction Distributed Systems and applications, Distributed vs parallel systems, models of distributed systems, Message Passing mechanisms IPC and RPC. | | |
| **Module2** | | **12 Hours** |
| Clock synchronization, physical & logical clocks, vector clocks, verifying clock algorithms, mutual exclusion using time stamp, election algorithms, Distributed mutual exclusion using time stamps, token & quorums, centralized & distributed algorithms, proof of correctness & complexity, drinking philosophers problem, Implementation & performance evaluation of DME Algorithms. | | |
| **Module 3** | | **11 Hours** |
| Leader election algorithms, global states, global predicates, termination detection, Control of distributed computation, disjunctive predicates, performance evaluation of leader election algorithms on simulated environments. | | |
| **Module 4** | | **12 Hours** |
| Distributed File Systems and Services, Shared data, Synchronization Transaction and Concurrency Control. Distributed databases, Name service, Timing & Coordination, Replication, Security and Fault Tolerance. | | |
| **Reference Books** | 1. Vijay K Garg, *Elements of Distributed Computing*, Wiley & Sons, 2002.<br>2. Pradeep Sinha, *Distributed Operating Systems- Concepts and Design*, PHI, 2000.<br>3. A.S. Tanenbaum, M.V. Steen, *Distributed Systems – Principles and Paradigms*, PHI, 2003<br>4. George Couloris, Jean Dollimore, Time Kindberg, *Distributed Systems: Concepts & Design*", Addison Wesley, 2003.<br>5. Nancy Lynch, *Distributed Algorithm*, Morgan Kaufmann Publishers, 1996. | |

| Subject Code CS804 | **Design of Secure Protocols (DSP)** | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives** | In this course, we investigate the paradigm of practice-oriented provable security in the context of public key cryptography. Central to this paradigm is the notion of security definition of a cryptographic task. Next comes the problem of designing protocols that can be proven secure assuming the intractability of certain computational problem(s) or the security of some atomic primitive(s). Several such cryptographic protocols will be studied in the course. | |
| **Module 1** | | **8 Hours** |
| Introduction to Cryptography: Basics of Symmetric Key Cryptography, Basics of Assymetric Key Cryptography, Hardness of Functions . One-way functions, one-way trapdoor functions. Notions of Semantic Security (SS) and Message Indistinguishability (MI): Proof of Equivalence of SS and MI, Hard Core Predicate, Trap-door permutation. | | |
| **Module 2** | | **6 Hours** |
| Formal Notions of Attacks: Attacks under Message Indistinguishability: Chosen Plaintext Attack (IND-CPA), Chosen Ciphertext Attacks (IND-CCA1 and IND-CCA2), Attacks under Message Non-malleability: NM-CPA and NM-CCA2, Inter-relations among the attack model. Random Oracles: Provable Security and asymmetric cryptography, hash functions One-way functions: Weak and Strong one way functions | | |
| **Module 3** | | **9 Hours** |
| Provably secure Pseudo-random Generators (PRG): Blum-Micali-Yao Construction, Construction of more powerful PRG, Relation between One-way functions and PRG, Pseudo-random Functions (PRF). Building a Pseudorandom Permutation. Provable security under different attacks of block ciphers, stream ciphers. Symmetric Encryption. | | |
| **Module 4** | | **10 Hours** |
| Message authentication: MAC, Authenticated encryption. Public key encryption: the notions of indistinguishability and semantic security including the question of equivalence of definitions, security against chosen plaintext and chosen ciphertext attacks. Some concrete public key encryption and identity-based encryption schemes and their security. | | |
| **Module 5** | | **12 Hours** |
| Digital signatures and the notion of existential unforgability under chosen message attacks. Key agreement protocols and secure channels. The random oracle assumption. The quantitative measure of security including the questions of tightness in security reduction and concrete security. Shamir's Secret Sharing Scheme, Formally Analyzing Cryptographic Protocols. Case Studies. | | |

| | |
|---|---|
| **Reference Books** | 1. Hans Delfs, Helmut Knebl, *Introduction to Cryptography: Principles and Applications*, Springer, 2002.<br>2. Wenbo Mao, *Modern Cryptography, Theory and Practice*, Prentice Hall, 2003.<br>3. Oded Goldreich, *Foundations of Cryptography,* Cambridge University Press, Vol-I and Vol-II, 2007.<br>4. Shaffi Goldwasser and Mihir Bellare, *Lecture Notes on Cryptography,* Available at http://citeseerx.ist.psu.edu.<br>5. Jonathan Katz, Yehuda Lindell, *Introduction to Modern Cryptography: Principles and Protocols*, Chapman & Hall/CRC Cryptography and Network Security Series, 2007. |

| Subject Code<br><br>CS805 | **Mobile Computing (MC)** | **Credits: 3 (3-0-0)**<br><br>**Total hours: 45** |
|---|---|---|
| **Course Objectives:** | This course briefly introduces the basic concepts, principles and developments in mobile computing. This includes major mobile communication technologies, mobile computing algorithms and support for mobility in current communication systems and Internet. | |
| **Module 1** | | **10 Hours** |
| History of wireless communications, market for mobile communications, open research topics, simplified reference model, wireless transmission technologies – frequencies for radio transmission, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, cellular networks. | | |
| **Module 2** | | **11 Hours** |
| Medium access control – techniques and algorithms, telecommunication systems – GSM, GPRS, DECT, TETRA, UMTS, CDMA, 3G, satellite systems – GEO, LEO, MEO, routing, localization, handover, wireless LAN – IEEE 802.11, HIPERLAN, Bluetooth. | | |
| **Module 3** | | **12 Hours** |
| Mobile network layer – Mobile IP, DHCP, mobile ad-hoc networks, mobile transport layer – indirect TCP, snooping TCP, mobile TCP, security issues in mobile computing. | | |
| **Module 4** | | **12 Hours** |
| Support for mobility in current communication systems and Internet – wireless application protocol, file systems, mobile web applications, mobile native applications, web 2.0, Voice over IP. | | |
| **Reference Books** | 1. Jochen Schiller, *Mobile Communications*, Pearson Education Limited, 2003.<br>2. Roopa Yavagal, Asoke K Talukder, *Mobile Computing – Technology, Applications and Service,* McGraw-Hill Professional, 2006. | |

| Subject Code CS806 | **Machine Learning (ML)** | **Credits: 3 (3-0-0) Total hours: 45** |
|---|---|---|
| **Course Objectives** | Machine learning is concerned with the question of how to make computers learn from experience. Machine learning techniques are used to create spam filters, to analyze customer purchase data, to understand natural language, or to detect fraudulent credit card transactions. This course will introduce the fundamental set of techniques and algorithms that constitute machine learning as of today, ranging from classification methods like decision trees and support vector machines, over structured models like hidden Markov models, to clustering and matrix factorization methods for recommendation. | |
| **Module 1** | | **8 Hours** |
| Algorithmic models of learning. Learning classifiers, functions, relations, grammars, probabilistic models, value functions, behaviors and programs from experience. Bayesian, maximum a posteriori, and minimum description length frameworks. | | |
| **Module 2** | | **12 Hours** |
| Parameter estimation, sufficient statistics, decision trees, neural networks, support vector machines, Bayesian networks, bag of words classifiers, N-gram models; Markov and Hidden Markov models, probabilistic relational models, association rules, nearest neighbor classifiers, locally weighted regression, ensemble classifiers. | | |
| **Module 3** | | **14 Hours** |
| Computational learning theory, mistake bound analysis, sample complexity analysis, VC dimension, Occam learning, accuracy and confidence boosting. Dimensionality reduction, feature selection and visualization. Clustering, mixture models, k-means clustering, hierarchical clustering, distributional clustering. | | |
| **Module 4** | | **11 Hours** |
| Reinforcement learning; Learning from heterogeneous, distributed, data and knowledge. Selected applications in data mining, automated knowledge acquisition, pattern recognition, program synthesis, text and language processing, internet-based information systems, human-computer interaction, semantic web, and bioinformatics and computational biology. | | |
| **Reference Books** | 1. Bishop, C., *Pattern Recognition and Machine Learning*, Berlin: Springer-Verlag, 2006.<br>2. Tom Mitchell, *Machine Learning*, McGraw Hill, 1997.<br>3. Hastie, Tibshirani, Friedman, *The Elements of Statistical Learning*, Springer, 2001.<br>4. Sergios Theodoridis, Konstantinos Koutroumbas, *Pattern Recognition*, Academic Press, 2009. | |

| Subject Code<br>CS807 | **Health Informatics (HI)** | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives:** | This course introduces the field of health informatics - which is an intersection of biomedical science, patient care, public health and information technology. | |
| **Module 1** | | **10 Hours** |
| Overview of health informatics, computer architectures and software engineering for healthcare and biomedicine, standards in health informatics. | | |
| **Module 2** | | **12 Hours** |
| Healthcare data, information and knowledge, health information exchange, health information security, health information infrastructure, biomedical decision making, introduction to bioinformatics. | | |
| **Module 3** | | **12 Hours** |
| Electronic health record systems, telemedicine, patient monitoring systems, public health informatics, patient-centered care systems. | | |
| **Module 4** | | **11 Hours** |
| Evidence-based medicine and clinical practice guidelines, ethics in health informatics, health information technology policy, future of health informatics. | | |
| **Reference Books** | 1. Edward H. Shortliffe, James J. Cimino, *Biomedical Informatics: Computer Applications in Health Care and Biomedicine* , Springer, 2012.<br>2. Robert E Hoyt , Nora Bailey, Ann Yoshihashi, *Health Informatics: Practical Guide For Healthcare And Information Technology Professionals,* lulu.com, 2012. | |

| **Subject Code CS808** | **Soft Computing (SC)** | **Credits: 3 (3-0-0) Total hours:45** |
|---|---|---|
| **Course Objectives** | To deal with the uncertainty that is inherent in any pattern recognition task. The uncertainty is natural in the real world also and human brain deals with it efficiently. | |
| **Module 1** | | **10 Hours** |
| Introduction to artificial neural networks (ANNs): artificial neuron as a computational model of a biological neuron, activation functions, learning laws, architectures for neural networks, Perceptron: learning law, convergence theorem. Multilayer feed forward neural networks: Structure, error back propagation learning, delta learning law, generalized delta rule, learning factors, convergence theorem, momentum factor in learning, conjugate based learning method, bias-variance dilemma | | |
| **Module 2** | | **8 Hours** |
| Deep learning: feedback neural networks, recurrent neural networks, convolution neural networks, Boltzan machine; Competitive learning models: principal component analysis, self-organizing map (SOM); Pulsed neural networks | | |
| **Module 3** | | **13 Hours** |
| Basic concepts of fuzzy logic: crisp set-properties, relations and operations, fuzzy set theory, membership, types of membership functions, uncertainty, fuzzification, Decision making using the fuzzy sets, fuzzy inference systems, defuzzification methods, Application of fuzzy systems, Introduction to Type-2 fuzzy logic systems: The structure, inference system with different fuzzy membership functions: Fuzzy clustering method: soft clustering, fuzzy K-means clustering method: Neuro-fuzzy systems: fuzzy logic with adaptive learning, adaptive neuro-fuzzy inference systems: Fuzzy-neuro systems: Fuzzy perceptron and learning method for the same, fuzzy back propagation network, | | |
| **Module 4** | | **14 Hours** |
| Evolutionary computing: optimization problem solving - finding best solution, minimum seeing algorithms, natural optimization methods, Genetic algorithms: Overview, a simple genetic algorithm, binary genetic algorithm, continuous parameter genetic algorithm. Advanced operations and techniques in genetic search, genetics-based machine learning – introduction and application, genetic algorithms in scientific methods. Genetic algorithms for combinatorial optimization, theoretical foundations of genetic algorithms, SASEGASA – parallel genetic algorithm. Introduction to genetic programming, applications of genetic programming, data-based modeling with genetic programming. Other evolutionary computing methods such as: ant colony optimization, swarm optimization. | | |
| **Reference Books** | 1. Satish Kumar, *Neural networks: A classroom approach*, Tata McGraw Hill, 2011.<br>2. B. Yegnanarayana, *Artificial Neural Networks*, Printice Hall India, 1999.<br>3. J. S. R. Lang, C. T. Sun and E. Mizutaju, *Neuro-fuzzy and soft computing*, Pearson Education, 1996.<br>4. David E. Goldberg , *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.<br>5. Michael Affenzeller, Stephan Winkler, Stefan Wagner, Andreas Beham, *Genetic Algorithms and Genetic Programming: Modern Concepts and Practical Applications*, CRC Press. 2009. | |

| Subject Code CS809 | **Service Oriented Architecture and Cloud Computing (SOAC)** | **Credits: 3 (3-0-0)**<br>**Total hours: 45** |
|---|---|---|
| **Course Objectives:** | This course introduces the fields of:<br>1) Service-oriented architecture (SOA) - which is a software architecture design pattern based on discrete pieces of software providing application functionality as services to other applications.<br>2) Cloud computing – which is a model for delivering services in which resources are retrieved from the internet through web-based tools and applications. | |
| **Module 1** | | **12 hours** |
| Introduction to SOA – fundamentals, characteristics, misperceptions, benefits and pitfalls, continuing evolution of SOA, principles of service orientation, service layers, Web services and SOA – web services framework, WSDL, SOAP, activity management and composition, advanced messaging, metadata and security. | | |
| **Module 2** | | **8 hours** |
| Planning and analysis for building SOA – SOA delivery lifecycle phases, service-oriented analysis, service modeling, service modeling applications, SOA patterns for performance, scalability and availability, security and manageability patterns, message exchange patterns, service consumer patterns, service integration patterns. | | |
| **Module 3** | | **10 hours** |
| Service-oriented design, services composition, service design guidelines, business process design, WS-BPEL, fundamental WS-* extensions, SOA platforms. | | |
| **Module 4** | | **15 hours** |
| Introduction to cloud computing, major models – software as a service, platform as a service, and infrastructure as a service, adopting SOA with cloud computing, data in the cloud – Cassandra, MangoDB, intelligence in the cloud, cloud security and governance. | | |
| **Reference Books** | 1. Thomas Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*, Prentice Hall, 2005.<br>2. Arnon Rotem-Gal-Oz, *SOA Patterns*, Manning Publications Company, 2012.<br>3. Michael Rosen, Boris Lublinsky, Kevin T. Smith, Marc J. Balcer, *Applied SOA: Service-Oriented Architecture and Design Strategies*, John Wiley & Sons, 2012.<br>4. Richard Hill, Laurie Hirsch, Peter Lake, Siavash Moshiri, *Guide to Cloud Computing: Principles and Practice*, Springer-Verlag, London, 2013.<br>5. Douglas K. Barry, *Web Services, Service-Oriented Architectures, and Cloud Computing, The Savvy Manager's Guide*, Morgan Kaufmann Publishers , 2nd Edition, 2003. | |

| Subject Code<br>CS810 | **Big Data Analytics (BDA)** | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives:** | Big data refers to a collection of large and complex data sets those are difficult to process using traditional data processing applications. The challenges include capture, curation, storage, search, sharing, transfer, analysis and visualization. This course introduces concepts and techniques to overcome these challenges and to infer laws from large data sets to reveal relationships, dependencies, and to perform predictions of outcomes and behaviors. | |
| **Module 1** | | **11 Hours** |
| Introduction – small and big data, statistics and machine learning, statistical data mining; Providing structure to unstructured data – machine translation, autocoding, indexing, term extraction; Identification, deidentification and reidentification; ontologies and semantics; introspection; data integration and software interoperability; immutability and immortality; measurement; big data and healthcare. | | |
| **Module 2** | | **12 Hours** |
| Big data techniques – data range, denominator, frequency distributions, mean and standard deviation, estimation-only analysis; big data analysis – clustering, classifying, recommending and modeling, data reduction, normalizing and adjusting data; special considerations – theory in search of data, data in search of a theory, overfitting, bigness bias, too much data, fixing data; stepwise approach to big data analysis – formulate a question, resource evaluation, reformulate a question, query output adequacy, data description and reduction, algorithm selection, results review; failure, legalities and societal issues. | | |
| **Module 3** | | **11 Hours** |
| Variable assessment - correlation coefficient, scatterplots; paired-variable assessment – CHAID based data mining; symmetrizing ranked data – scales of measurement, Stem-and-Leaf display, Box-and-Whiskers plot; many-variable assessment – principle component analysis; logistic regression; ordinary regression; regression coefficient; predictive contribution coefficient. | | |
| **Module 4** | | **11 Hours** |
| R language – data modeling in R, importing data into R, Hadoop – different Hadoop modes, Hadoop Distributed File System (HDFS) – fundamentals and architecture, MapReduce – fundamentals and architecture, Hadoop security, Hadoop programming in Java, Integrating R and Hadoop – RHIPE, RHadoop, data analytics with R and Hadoop, importing and exporting data from various databases, Hive, RBase, Apache Pig- large data analysis platform, automating data processing with Oozie. | | |
| **Reference Books** | 1. Jules J Berman, *Principles of Big Data: Preparing, Sharing, and Analyzing Complex Information*, Morgan Kaufman-Elsevier, 2013.<br>2. Bruce Ratner , *Statistical and Machine-Learning Data Mining: Techniques for Better Predictive Modeling and Analysis of Big Data*, 2nd Edition, CRC Press, 2011.<br>3. Michael Milton, *Head First Data Analysis: A learner's guide to big numbers, statistics, and good decisions*, O'Reilly Media Inc., 2009.<br>4. *Big Data Now: 2012 Edition*, O'reilly Media Inc., 2012.<br>5. Vignesh Prajapati , *Big Data Analytics with R and Hadoop*, Packt Publishing, 2013. | |

| | | |
|---|---|---|
| **Subject Code CS811** | **Pattern Recognition (PR)** | **Credits: 3 (3-0-0) Total hours:45** |
| **Course Objectives** | To build intelligent systems based on the learning framework. | |
| **Module 1** | | **12 Hours** |
| Pattern classification: Bayesian decision theory, minimum-error-rate classification, classifiers, discriminant functions, decision surfaces, normal (Gaussian) density, continuous and discrete values features, Bayesian networks (graphical models) | | |
| **Module 2** | | **8 Hours** |
| Methods for parameter estimation: maximum likelihood estimation, maximum a posteriori estimation, Bayesian estimation, Gaussian mixture models<br>Sequential pattern classification: Hidden Markov models for dynamic patterns | | |
| **Module 3** | | **10 Hours** |
| Non-parametric method for density estimation: Parzon window and K-nearest neighbor method<br>Methods for dimensionality reduction: Fisher's discriminant analysis, Principal component analysis<br>Non metric methods: Decision trees, classification and regression trees (CART), recognition of strings | | |
| **Module 4** | | **8 Hours** |
| Discriminant analysis: Models for decision surfaces, linear discriminant analysis-perception model, minimum mean squared error based learning, support vector machines<br>Repression: Linear models for regression, polynomial regression, Bayesian regression | | |
| **Module 5** | | **7 Hours** |
| Pattern clustering (unsupervised learning): Criterion functions for clustering, methods for clustering-hard and soft clustering, K-means, GMM, hierarchical clustering methods, cluster validation methods | | |
| **Reference Books** | 1. Richard O. Duda, Peter E. Hart and David G. Stork, *Pattern Classification*, 2nd Edition, John Wiley & Sons, 2012.<br>2. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.<br>3. Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition*, 4th Edition, Academic Press-Elsevier, 2009 | |

<table>
<tr><td>Subject Code<br>CS812</td><td>Artificial Neural Networks (ANN)</td><td>Credits: 3 (3-0-0)<br>Total hours:45</td></tr>
<tr><td>Course Objectives</td><td colspan="2">To study a computational model of the human neural system though it is still not known the exact functioning of the same.</td></tr>
<tr><td colspan="2">Module 1</td><td>8 Hours</td></tr>
<tr><td colspan="3">Biological neuron, artificial neuron as a computational model of a neuron, activation functions, architectures for ANNs, linear neural networks, Hebbs learning law,</td></tr>
<tr><td colspan="2">Module 2</td><td>14 Hours</td></tr>
<tr><td colspan="3">Non-linear neural networks: Perceptron- learning law, convergence theorem; multilayer feed forward neural networks-structure, activation functions, error back propagation learning, delta learning law, generalized delta rule, learning factors, convergence criteria, momentum factor in learning, conjugate gradient method for learning, universal approximation theorem, cross validation method for selecting the architecture, bias-variance dilemma</td></tr>
<tr><td colspan="2">Module 3</td><td>8 Hours</td></tr>
<tr><td colspan="3">Statistical learning theory, principle of empirical risk minimization, Radial basis function networks: RBF networks for function approximation, RBF networks for pattern classification, Support vetcor machines: SVM for linearly separable classes, SVM for linearly non-separable classes, SVM for nonlinearly separable classes using kernels, multi-class pattern classification using SVMs,</td></tr>
<tr><td colspan="2">Module 4</td><td>8 Hours</td></tr>
<tr><td colspan="3">Feedback neural networks: Problem of pattern storage and retrieval, discrete Hopfiled networks, dynamical systems, energy function of hopfield model, energy analysis of hopfiled model.</td></tr>
<tr><td colspan="2">Module 5</td><td>7 Hours</td></tr>
<tr><td colspan="3">Introduction to deep neural networks, convolution neural networks, recurrent neural networks, Boltzman machine.</td></tr>
<tr><td>Reference Books</td><td colspan="2">1. B. Yegnanarayana, Artificial Neural Networks, Printice Hall India Learning Pvt. Ltd, 2009.<br>2. Sathish Kumar, Neural Networks: A Classroom Approach, 3rd Edition, Tata McGraw Hill, 2011.<br>3. Simon S. Haykin, Neural Networks and Learning Machines, 3rd Edition, Prentice Hall, 2009</td></tr>
</table>

| Subject Code CS813 | **Computer Vision (CV)** | Credits: 3 (3-0-0) Total hours:45 |
|---|---|---|
| **Course Objectives** | To expose the students to fundamental and advanced topics in computer vision with a focus on image statistics, machine learning techniques, and applied vision for graphics also. | |
| **Module 1** | | **10 Hours** |
| Introduction and overview, pinhole cameras, radiometry terminology. Sources, shadows and shading: Local shading models- point, line and area sources; photometric stereo. Color: Physics of color; human color perception, Representing color; A model for image color; surface color from image color. | | |
| **Module 2** | | **13 Hours** |
| Image Processing: Linear filters: Linear filters and convolution; shift invariant linear systems- discrete convolution, continuous convolution, edge effects in discrete convolution; Spatial frequency and Fourier transforms; Sampling and aliasing; filters as templates; Normalized correlations and finding patterns. Edge detection: Noise; estimating derivatives; detecting edges. Texture: Representing texture; Analysis using oriented pyramid; Applications; Shape from texture. The geometry and views: Two views. | | |
| **Module 3** | | **12 Hours** |
| Stereopsis: Reconstruction; human stereo; Binocular fusion; using color camera. Segmentation by clustering: Human vision, applications, segmentation by graph theoretic clustering. Segmentation by fitting a model, Hough transform; fitting lines, fitting curves; | | |
| **Module 4** | | **10 Hours** |
| 3D reconstruction, model based vision- face recognition, face detection, image/scene classification, motion tracking, surveillance, content based image and video retrieval | | |
| **Reference Books** | 1. Richard Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2011.<br>2. David A Forsyth and Jean Ponce, *Computer Vision, A Modern Approach*, Pearson Education, Limited, 2011<br>3. Schalkoff R. J., *Digital Image Processing and Computer Vision*, John Wiley & Sons Australia, Limited, 1989<br>4. Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, 3rd Eidtion, Pearson Eductaion India, 2009<br>5. Milan Sonka, Vaclav Hlavac and Roger Boyle, *Image Processing, Analysis, and Machine Vision*, 4th Edition, Cengage Learning, 2014 | |

| Subject Code<br>CS814 | Game Theory (GMT) | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives** | Game theory and mechanism design offer an important tool to model, analyze, and solve decentralized design problems involving multiple autonomous agents that interact strategically in a rational and intelligent way. This course provides a sound foundation of game theory and mechanism design to enable the audience to apply them to problem solving in a rigorous way. | |
| **Module 1** | | **7 Hours** |
| Introduction and Outline of the Course, Definitions, Utilities, Rationality, Intelligence, Common Knowledge, Classification of Games. | | |
| **Module 2** | | **14 Hours** |
| **Non-Cooperative Game Theory**: Extensive Form Game, Strategic Form Games with Illustrative Examples, Dominant Strategy Equilibria, Pure Strategy Nash Equilibrium with Illustrative Examples and Key Results, Mixed Strategy Nash Equilibrium with Illustrative Examples and Key Results such as the Nash Theorem, Computation of Nash Equilibria and introduction to algorithmic theory, Matrix Games: Saddle Points, Minimax Theorem, Bayesian Games, Bayesian Nash Equilibrium, Evolutionary Game Theory (ESS Strategies), Repeated Game. | | |
| **Module 3** | | **12 Hours** |
| **Mechanism Design:** The Mechanism Design Environment, Social Choice Functions with Illustrative Examples , Implementation of Social Choice Functions, Incentive Compatibility and Revelation Theorem, Gibbard-Satterthwaite and Arrow Impossibility Theorem, Vickrey-Clarke-Groves (VCG) Mechanisms, Bayesian Mechanisms (dAGVA), Revenue Equivalence Theorem, Myerson Optimal Auction, Further Topics in Mechanism Design | | |
| **Module 4** | | **12 Hours** |
| Correlated Strategies and Correlated Equilibrium, The Nash Bargaining Problem, Coalitional Games (Transferable Utility Games), The Core, The Shapley Value, Other Solution Concepts: Kernel, Nucleolus. | | |
| **Reference Books** | 1. Martin J. Osborne, *An Introduction to Game Theory*, Oxford University Press, 2009.<br>2. Roger B. Myerson, *Game Theory: Analysis of Conflict,* Harvard University Press, 1997.<br>3. Y. Narahari, Dinesh Garg, Ramasuri Narayanam, and Hastagiri Prakash, *Game Theoretic Problems in Network Economics and Mechanism Design Solutions,* Springer, London, 2009. | |

| Subject Code<br>CS815 | **Data Warehousing and Data Mining (DWM)** | Credits: 3 ( 3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives** | Following this course, students will be able to 1) Learn the concepts of database technology, 2) Understand data mining principles and techniques, 3) Discover interesting patterns from large amounts of data to analyze and extract patterns to solve problems, make predictions of outcomes. 4) Evaluate systematically supervised and unsupervised models and algorithms with respect to their accuracy, 5) Design and implement of a data-mining application using sample, realistic data sets and modern tools. | |
| **Module 1** | | **12 Hours** |
| Introduction to data warehousing, building a data warehouse, mapping the data warehouse to a multiprocessor architecture, OLAP technology for data mining, data warehouse, multidimensional data model, data warehouse architecture, data warehouse implementation, OLAP guidelines, multidimensional versus multi relational OLAP, categories of tools, DBMS schemas for decision support data extraction, cleanup and transformation tools for metadata, development of data cube technology, from data warehousing to data mining, data generalization, efficient methods for data cube computation, further development of data cube and OLAP Technology, attribute-oriented induction. | | |
| **Module 2** | | **8 Hours** |
| Introduction to data mining tasks, objectives (classification, clustering, association rules, sequential patterns, regression, deviation detection). | | |
| **Module 3** | | **8 Hours** |
| Data and preprocessing (data cleaning, feature selection, dimensionality reduction), Curse of Dimensionality | | |
| **Module 4** | | **8 Hours** |
| Classification (decision-tree based approach, rule-based approach, instance-based classifiers, Bayesian Approach: Naive and Bayesian networks, classification model evaluation). | | |
| **Module 5** | | **9 Hours** |
| Clustering (partitional methods, hierarchical methods, graph-based methods, density-based methods, cluster validation methods), anomaly/outlier detection (introduction to various types of outliers, statistical-based, density-based and other methods for outlier detection). | | |
| **Reference Books** | 1. Jiawei Han and Micheline Kamber, *Data mining: Concepts and techniques*, 3rd Edition, Morgan Kaufmann publishers, 2012.<br>2. Raph Kimball and Margy Ross, *Data warehouse toolkit*, 3rd Edition, John Wiley & Sons Publications, 2013.<br>3. Gordon Linoff and Michael. J. Berry, *Data mining techniques: Marketing, sales, customer support*, 3rd Edition, John Wiley & Sons, 2011. | |

| **Subject Code CS816** | **E-Commerce (EC)** | **Credits: 3 (3-0-0) Total hours: 45** |
|---|---|---|
| **Course Objectives** | To provide principles of e-commerce from a business perspective. | |
| **Module 1** | | **11 Hours** |
| Infrastructure and tools for e-commerce, current trends in e-commerce applications development, the business of internet commerce, enterprise level e-commerce. | | |
| **Module 2** | | **12 Hours** |
| Security and encryption, electronic payment systems, search engines, intelligent agents in e-commerce, on-line auctions, data mining for e-commerce. | | |
| **Module 3** | | **12 Hours** |
| Web metrics, recommended systems, knowledge management, mobile e-commerce, legal, ethical and social issues. | | |
| **Module 4** | | **10 Hours** |
| Seminars and mini projects. | | |
| **Reference Books** | 1. Henry Chan, Raymond Lee, Tharam Dillon and Elizabeth Chang, *E-Commerce-Fundamentals and application*, John Wiley & Sons 2007.<br>2. G. Winfield Treese and Lawrence C. Stewart, *Designing Systems for Internet Commerce*, Addison-Wesley Professional, 2003.<br>3. M. L. Brodie and Dieter Fensel, *Ontologies: A Silver Bullet for Knowledge Management and ECommerce*, Springer, 2004.<br>4. Olaf Zimmermann, Mark Tomlinson and Stefan Peuser, *Perspectives on Web Services*", Springer, 2004. | |

<table>
<tr><td>Subject Code<br><br>CS817</td><td colspan="2">Advanced Operating Systems (AOS)</td><td>Credits: 3 (3-0-0)<br><br>Total hours: 45</td></tr>
<tr><td>Course Objectives</td><td colspan="3">To provide comprehensive and up-to-date coverage of the major developments in distributed operating system, multi-processor operating system and database operating system.</td></tr>
<tr><td colspan="3">Module 1</td><td>8 Hours</td></tr>
<tr><td colspan="4">Architectures of distributed systems , system architecture types, issues in distributed OS, communication networks, primitives, theoretical foundations, inherent limitations of a distributed system, lamp ports logical clocks, vector clocks, casual ordering of messages, global state, cuts of a distributed computation, termination detection, distributed mutual exclusion.</td></tr>
<tr><td colspan="3">Module 2</td><td>12 Hours</td></tr>
<tr><td colspan="4">Distributed deadlock detection, introduction, deadlock handling strategies in distributed systems, issues in deadlock detection and resolution, control organizations for distributed deadlock detection, centralized, distributed and hierarchical deadlock detection algorithms , agreement protocols.</td></tr>
<tr><td colspan="3">Module 3</td><td>10 Hours</td></tr>
<tr><td colspan="4">Distributed shared memory, architecture, algorithms for implementing DSM, memory coherence and protocols, design issues, distributed scheduling, issues in load distributing, components of a load distributing algorithm, stability, load distributing algorithm, performance comparison, selecting a suitable load sharing algorithm, requirements for load distributing, task migration and associated issues. Failure recovery and Fault tolerance: Introduction, basic concepts, classification of failures, backward and forward error recovery, recovery in concurrent systems, consistent set of check points, synchronous and asynchronous check pointing and recovery, check pointing for distributed database systems, recovery in replicated distributed databases.</td></tr>
<tr><td colspan="3">Module 4</td><td>8 Hours</td></tr>
<tr><td colspan="4">Protection and security, preliminaries, the access matrix model and its implementations, safety in matrix model, advanced models of protection. Cryptography basics, multiple encryption and authentication in distributed systems.</td></tr>
<tr><td colspan="3">Module 5</td><td>7 Hours</td></tr>
<tr><td colspan="4">Multiprocessor OS, database OS, database systems, a concurrency control model, problem, serializability theory, distributed database systems, concurrency control algorithms.</td></tr>
<tr><td>Reference Books</td><td colspan="3">1. Mukesh Singhal and Niranjan G. Shivorothri, Advanced Concepts in Operating systems: Distributed, Multiprocessor and Database Operating Systems, McGraw-Hill Education, 1994.<br>2. Andrew S. Tanenbaum, Distributed Operating systems, Pearson Education, 2008.<br>3. Doreen L. Galli, Distributed operating systems: concepts and practice, Prentice Hall, 2000.<br>4. Abraham Silberschatz and Avi Silberschatz, Applied Operating systems, John Wiley & Sons, 2000.<br>5. Lubomir F. Bic and Alan C. Shaw, Operating systems Principles, Prentice Hall PTR, 2003.</td></tr>
</table>

<table>
<tr><td>Subject Code<br>CS818</td><td colspan="2">Security and Privacy (S&P)</td><td>Credits: 3 (3-0-0)<br>Total hours: 45</td></tr>
<tr><td>Course Objectives</td><td colspan="3">This course introduces the concepts of security and privacy.</td></tr>
<tr><td colspan="3">Module 1</td><td>10 Hours</td></tr>
<tr><td colspan="4">Introduction: Basic concepts: number theory, Formal analysis and design of algorithms and protocols.</td></tr>
<tr><td colspan="3">Module2</td><td>10 Hours</td></tr>
<tr><td colspan="4">Provable Security, Cryptosystems; Privacy: Foundations of Privacy, Differential Privacy: Definitions and Early Uses.</td></tr>
<tr><td colspan="3">Module 3</td><td>10 Hours</td></tr>
<tr><td colspan="4">Privacy Regulations, Noiseless Differential Privacy, Privacy preserving Data Mining techniques.</td></tr>
<tr><td colspan="3">Module 4</td><td>15 Hours</td></tr>
<tr><td colspan="4">Privacy preserving data publishing: Fundamental Concepts: anonymization methods, privacy models, anonymization method for trasaction data, trajectory data, social networks data and textual data. One-Time Data Publishing, Multiple-Time Data Publishing :Graph Data .Other Data Types . Access control of outsourced data. Future Research Directions</td></tr>
<tr><td>Reference Books</td><td colspan="3">1. T. Shaw, Information Security and Privacy, American Bar Association, 2012.<br>2. M. Bailey, Complete Guide to Internet Privacy, Anonymity and Security, Nerel Online, 2011.<br>3. Raymond Chi-Wing Wong, Ada Wai-Chee Fu, Privacy-Preserving Data Publishing: An Overview, Morgan and claypool publishers, 2010.</td></tr>
</table>

| Subject Code<br>CS819 | **Bioinformatics Algorithms (BA)** | Credits: 3 (3-0-0)<br>Total hours:45 |
|---|---|---|
| **Course Objectives** | To explore fundamental algorithmic techniques in bioinformatics and computational biology that are enabling the current revolution in life sciences and medicine It will serve as the foundation course for students of computer science who are interested in doing research or pursue career in computational biology or in bioinformatics. | |
| **Module 1** | | |
| Introduction to molecular biology – Basic introduction including DNA, proteins, central dogma etc., what is it involved in analyzing a DNA, role of bioinformatics | | |
| **Module 2** | | |
| Pairwise sequence alignments – Global, semi-global and local alignments, gap penalty functions, Hirshberg's space-saving algorithm, banded dynamic programming. Multiple sequence alignments – sum-of-pairs scoring function, Carillo-Lippman heuristic, approximation algorithms, tree alignments. | | |
| **Module 3** | | |
| String data structures and algorithms – look-up tables, suffix arrays and suffix trees, construction algorithms, basic applications of suffix trees, lowest common ancestors. | | |
| **Module 4** | | |
| Genome assembly – overlap-layout-consensus and graph based methods. Comparative genomics – Identifying gene clusters and evolutionarily conserved sequences. Pairwise and multiple genome comparisons. | | |
| **Module 5** | | |
| Phylogenetics – distance based methods including ultrametric and additive distances, character based methods including parsimony and perfect phylogeny, heuristic methods. | | |
| **Reference Books** | 1. N. C. Jones & P. A. Pevzner, *An Introduction to Bioinformatics Algorithms*, MIT Press, 2004.<br>2. R. Durbin, S. Eddy, A. Krogh, G. Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids,* Cambridge University Press, 1998.<br>3. S. Aluru, *Handbook of computational molecular biology*, Chapman and Hall/CRC, 2005. | |

| Subject Code<br><br>CS820 | **Graph Theory (GT)** | | **Credits: 3 (3-0-0)**<br><br>**Total hours: 45** |
|---|---|---|---|
| **Course Objectives** | The intension of this course is to introduce the subject of graph theory to computer science students in a thorough way. While the course will cover all elementary concepts such as coloring, covering, Hamiltonicity, planarity, connectivity and so on, it will also introduce the students to some advanced concepts. | | |
| **Module 1** | | | **8 Hours** |
| Definitions, pictorial representation of a graph, isomorphic graphs, sub graphs, matrix representations of graphs, degree of a vertex, special graphs, complements, larger graphs from smaller graphs, connected graphs and shortest paths, walks, trails, paths, cycles, connected graphs, cut-vertices and cut-edges, blocks, connectivity, weighted graphs and shortest paths, weighted graphs, Dijkstra's shortest path algorithm, Floyd-Warshall shortest path algorithm | | | |
| **Module2** | | | **8 Hours** |
| Trees, Definitions and characterizations, number of trees, Cayley's formula, minimum spanning trees, Kruskal's algorithm, Prim's algorithm, bipartite graphs, Eulerian graphs, Fleury's algorithm, Chinese Postman problem. . | | | |
| **Module 3** | | | **8 Hours** |
| Hamilton Graphs, necessary conditions and sufficient conditions, independent sets, coverings and matchings, matchings in bipartite graphs, Hall's theorem, Konig's theorem, perfect matching's in graphs, vertex Colorings, basic definitions, cliques and chromatic number, greedy coloring algorithm. | | | |
| Module 4 | | | **9 Hours** |
| Edge colorings, Gupta-Vizing theorem, class-1 and class-2 graphs, edge-coloring of bipartite, graphs, planar graphs, basic concepts, Euler's formula and its consequences, characterizations of planar graphs, 5-color-theorem, directed graphs, directed walks, paths and cycles, Eulerian and Hamilton digraphs. | | | |
| **Module 5** | | | **12 Hours** |
| Planarity (duality, Euler's formula, characterization, 4-color theorem); Advanced topics (perfect graphs, matroids, Ramsay theory, extremal graphs, random graphs); Applications. | | | |
| **Reference Books** | 1. D. B. West, *Introduction to Graph Theory*, 2nd edition, Prentice Hall, 2000.<br>2. R. Diestel, *Graph Theory (Graduate Texts in Mathematics)*, 2nd edition, Springer-Verlag, 2000.<br>3. J.A. Bondy and U.S.R. Murty, *Graph Theory (Graduate Texts in Mathematics)*, Springer, 2011.<br>4. R. P. Grimaldi, *Discrete and Combinatorial Mathematics: An Applied Introduction*, 5th edition, Pearson Education, Asia, 2003.<br>5. N. Alon and J. Spenser, *The Probabilistic Method*, 3rd edition, John Wiley and Sons, 2008. | | |

| Subject Code<br><br>**CS821** | **Probability and Statistics (P&S)** | **Credits: 3 (3-0-0)**<br><br>**Total hours: 45** |
|---|---|---|
| **Course Objectives** | Probability and Statistics is one of the most important branches of the mathematical sciences. Knowledge of these topics is critical to decision-making and to the analysis of data. Using concepts of probability and statistics, individuals are able to predict the likelihood of an event occurring, organize and evaluate data, and identify the significance of statements. | |
| **Module 1** | | **8 Hours** |
| **Algebra of Sets:** sets and classes, limit of a sequence of sets, rings, sigma-rings, fields, sigma-fields, monotone classes. **Probability:** Classical, relative frequency and axiomatic definitions of probability, addition rule and conditional probability, multiplication rule, total probability, Bayes' Theorem and independence, problems. | | |
| **Module2** | | **8 Hours** |
| **Random Variables:** Discrete, continuous, mixed random variables, probability mass, density and cumulative distribution functions, mathematical expectation, moments, probability and moment generating function, median and quantiles, Markov inequality, Chebyshev's inequality, problems. | | |
| **Module 3** | | **12 Hours** |
| **Distributions: (Special)** Discrete uniform, binomial, geometric, negative binomial, hypergeometric, Poisson, continuous uniform, exponential, gamma, Weibull, Pareto, beta, normal, lognormal, inverse Gaussian, Cauchy, double exponential distributions, reliability and hazard rate, reliability of series and parallel systems, problems. (**Joint**)**:** Joint, marginal and conditional distributions, product moments, correlation and regression, independence of random variables, bivariate normal distribution, problems. (**Sampling**)**:** The Central Limit Theorem, distributions of the sample mean and the sample variance for a normal population, Chi-Square, t and F distributions, problems. | | |
| **Module 4** | | **10 Hours** |
| **Estimation:** Unbiasedness, consistency, the method of moments and the method of maximum likelihood estimation, confidence intervals for parameters in one sample and two sample problems of normal populations, confidence intervals for proportions. **Testing of Hypotheses:** Null and alternative hypotheses, the critical and acceptance regions, two types of error, power of the test, the most powerful test, Neyman-Pearson Fundamental Lemma, tests for one sample, two sample problems for normal populations, tests for proportions, Chi-square goodness of fit test, applications. | | |
| **Module 5** | | **7 Hours** |
| **Transformations**: functions of random vectors, distributions of order statistics, distributions of sums of random variables, problems. | | |
| **Reference Books** | 1. V.K. Rohatgi, A.K. Md. E. Saleh, *An Introduction to Probability & Statistics*, 2nd edition, Wiley-Interscience, 2000.<br>2. J.S. Milton & J.C. Arnold, *Introduction to Probability and Statistics - Principles and Applications for Engineering and the Computing Sciences*, 4th edition, McGraw-Hill Higher Education, 2002.<br>3. H.J. Larson, *Introduction to Probability Theory and Statistical Inference*, 3rd edition, Wiley, 1982.<br>4. S.M. Ross, *Introduction to Probability & Statistics for Engineers and Scientists*, $4^{th}$ edition, Elsevier AP, 2009.<br>5. S.M. Ross, *A First Course in Probability*, $9^{th}$ edition, Pearson, 2012. | |

| Subject Code<br><br>CS822 | **Program Analysis and Verification (PAV)** | Credits: 3 (3-0-0)<br><br>Total hours: 45 |
|---|---|---|
| **Course Objectives:** | This course teaches techniques for model checking - a formal verification technique for assessing functional properties of information and communication systems. Model checking is an automated technique to check the absence of errors and it is considered as an intelligent and effective debugging technique. | |
| **Module 1** | | **9 hours** |
| Introduction to code verification, the mathematical model and numerical algorithm, the order-verification procedure and its benefits, design of coverage test suite, finding exact solutions, numerical algorithm development, testing for code robustness and code efficiency, dealing with codes that make non-ordered approximations. | | |
| **Module 2** | | **11 hours** |
| The semantic analysis – the precondition, the post condition, the principles of top-down refinement, program correctness – programs without loops, iterative programs, program test for any implementation – black box testing, static analysis – intermediate program representation, program dependencies, tell about a program without its execution, dynamic analysis – structural program testing, dynamic program analysis. | | |
| **Module 3** | | **11 hours** |
| System verification – model checking, modeling concurrent systems, linear-time properties, regular properties, linear temporal logic, computation tree logic, equivalences and abstraction, partial order reduction, timed automata, probabilistic systems – Markov chain and Markov decision processes. | | |
| **Module 4** | | **14 hours** |
| Finding bugs in concurrent systems, building verification models, an overview of PROMELA, defining correctness claims, using design abstraction, automata and logic, PROMELA semantics, search algorithms and optimization, model abstraction, using SPIN and XSPIN – the TimeLine editor, a verification model of a telephone switch, sample SPIN models. | | |
| **Reference Books** | 1. P. Knupp, K. Salari, *Verification of Computer Codes in Computational Science and Engineering*, Chapman & Hall/CRC, 2002.<br>2. J. Laski, W. Stanley, *Software Verification and Analysis: An Integrated, Hands-On Approach*, Springer, 2009.<br>3. B. Berard, M. Bidoit, A. Finkel, F. Laroussinie, A. Petit, L. Petrucci, P. Schnoebelen, P. McKenzie, *Systems and Software Verification: Model-Checking Techniques and Tools*, Springer, 2001.<br>4. G. Holzmann, *The SPIN Model Checker: Primer and Reference Manual*, Addison-Wesley, 2003.<br>5. C. Baier, J. P. Katoen, K. G. Larsen, *Principles of Model Checking*, MIT Press, 2008. | |

| Subject Code<br>CS823 | Linear Algebra (LA) | Credits: 3 (3-0-0)<br>Total hours: 45 |
|---|---|---|
| **Course Objectives** | To have a hand on in linear algebra to understand matrices and use them to various engineering applications. | |
| **Module 1** | | **9 hours** |
| Introduction to vectors: Vectors and linear combinations, dot products, matrices.<br>Solving linear equations: Vectors and linear equations, idea of elimination, eliminations using matrices, matrix operations, inverse of a matrix, LU and LDU factorizations, transposes and permutations | | |
| **Module 2** | | **9 hours** |
| Vector spaces and subspaces: The null subspace of A: Solving Ax=0, the rank and row reduced form, basis and dimension, four fundamental subspaces | | |
| **Module 3** | | **9 hours** |
| Orthogonality: Projections, least squares approximations, orthogonal bases and Gram-Schmidt<br>Determinants: Properties of determinants, Formulas for determinants, applications of determinants | | |
| **Module 4** | | **9 hours** |
| Eigen values and Eigen vectors: Introduction to Eigen values and Eigen vectors, diagonalization of a matrix, differential equations, symmetric matrices, positive definite matrices, | | |
| **Module 5** | | **9 hours** |
| Applications: Matrices in engineering, graphs and networks, Markov matrices linear programming, Fourier series, computer graphics, Gaussian elimination in practice | | |
| **Reference Books** | 1. G. Strang, *Introduction to Linear Algebra*, 4th Edition, Wellesley-Cambridge Press, Wellesley, MA, 2009.<br>2. G. Strang, *Linear algebra and its applications,* Thomson Books, 2006. | |

| Subject Code CS824 | Number Theory (NT) | Credits: 3 (3-0-0) Total hours: 45 |
|---|---|---|
| **Course Objectives** | This course introduces the number theory, Algebraic structures and the computational aspects of number theory. | |
| **Module 1** | | **8 Hours** |
| Preliminaries: Well ordering principle, Mathematical Induction. Divisibility Theory in Integers: Divisibility properties, Division Theorem, greatest common Divisor, Euclidean algorithm, Diophantine equation. Primes and their distribution: The fundamental theorem of arithmetic. | | |
| **Module2** | | **15 Hours** |
| Theory of Congruences: Basic properties of congruences, Divisibility tests, Linear congruences, Chinese Remainder Theorem, Fermat's theorem, Euler's theorem, Quadratic Residues and Reciprocity. Arithmetic Functions, Diophantine Equations. | | |
| **Module 3** | | **10 Hours** |
| Groups, Rings, Finite fields, Elliptic Curves, Elliptic Curve arithmetic | | |
| **Module 4** | | **12 Hours** |
| Large integer computations: Computations in $Z_n$ ; Primality testing of Integers; Integer Factorization algorithms. Computations in groups, Rings and Fields. Algorithms for discrete logarithms ; Polynomial arithmetic ; Sequence generation; Algorithms for Finite fields. | | |
| **Reference Books** | 1. N. Koblitz, *A Course in Number theory and Cryptography*, 2nd edition, Springer, 1994.<br>2. V. Shoup, *A Computational Introduction to Number Theory and Algebra*, Cambridge Press, 2008.<br>3. H. Cohen, *A course in Computational algebraic number theory,* 4th printing, Springer, 2000.<br>4. R. Lidle, H. Niderreiter, *Finite Fields (Encyclopedia of Mathematics and its Applications)*, 2nd edition, Cambridge University press, 2008. | |

| | | |
|---|---|---|
| **Subject Code**<br><br>**CS825** | **Complexity Theory (CT)** | **Credits: 3 (3-0-0)**<br><br>**Total hours: 45** |
| **Course Objectives** | This course introduces computational complexity theory. | |
| **Module 1** | | **10 Hours** |
| Fundamental concepts: problems and algorithms, Turing machines, computability. | | |
| **Module2** | | **12 Hours** |
| Complexity Classes: P, NP and co-NP, Relationship between complexity classes, Reduction and completeness, NP-complete problems, P vs NP. | | |
| **Module 3** | | **12 Hours** |
| Diagonalization and Relativization. Space complexity: PSPACE and PSPACE-completeness; NL and NL-completeness. The polynomial hierarchy: optimization problems. Non-uniform complexity. Communication complexity and circuit lower bounds. | | |
| **Module 4** | | **11 Hours** |
| Randomized computation: RP, BPP, ZPP. Error reduction. Probabilistic algorithms. Randomized space complexity. Approximation and Inapproximability. Interactive proofs. | | |
| **Reference Books** | 1. S. Arora and B. Barak, *Computational Complexity: A Modern Approach*, Cambridge University Press, 2009,<br>2. C. H. Papadimitriou, *Computational Complexity*, 1st edition, Addison Wesley, 1993. | |

<table>
<tr><td>Subject Code<br><br>CS826</td><td colspan="2">Human Computer Interface (HCI)</td><td>Credits: 3 (3-0-0)<br><br>Total hours: 45</td></tr>
<tr><td>Course Objectives:</td><td colspan="3">Human-Computer Interface (HCI) refers to the design, prototyping, and evaluation of user interfaces to computers. The following topics are covered by this course: human capabilities, interface technology, interface design methods and interface evaluation.</td></tr>
<tr><td colspan="3">Module 1</td><td>11 hours</td></tr>
<tr><td colspan="4">Human perception, human vision, Gestalt principles describing human visual perception, visual structure, color vision, peripheral vision, attention and memory, attention shape thought and action, recognition and recall, learning from experience, performing learned actions, problem solving and calculation, factors affecting learning, time requirements.</td></tr>
<tr><td colspan="3">Module 2</td><td>12 hours</td></tr>
<tr><td colspan="4">Usability of interactive systems – guidelines, principles and theories; development processes – managing design processes, evaluating interface designs, software tools; interaction styles – direct manipulation and virtual environments, menu selection, form filling and dialog boxes, command and natural languages, interaction devices, collaboration; design issues – Quality of Service, balancing function and fashion, user manuals, online help and tutorials, information search and visualization; societal and individual impact of user interfaces.</td></tr>
<tr><td colspan="3">Module 3</td><td>11 hours</td></tr>
<tr><td colspan="4">Interaction design – introduction, the process, the user experience; understanding and conceptualizing interaction – conceptual models, interface metaphors, interaction types; cognitive aspects – cognition frameworks; social interaction; emotional interaction; interfaces – interface types, natural user interfaces, choosing interface.</td></tr>
<tr><td colspan="3">Module 4</td><td>11 hours</td></tr>
<tr><td colspan="4">Data gathering; data analysis, interpretation and presentation; practical issues in the process of interaction design; establishing requirements – data gathering and processing, task description, task analysis; design, prototyping and construction – conceptual design, physical design, scenarios, prototypes; evaluation – types of evaluation, evaluation case studies, inspections – heuristic evaluation and walkthroughs, analytics, predictive models; evaluation framework – DECIDE – a framework to guide evaluation.</td></tr>
<tr><td>Reference Books</td><td colspan="3">1. B. Shneiderman, C. Plaisant, M. Cohen and S. Jacobs, Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th Edition, Person Education, 2009.<br>2. J. Johnson, Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules, Elsevier/Morgan-Kaufmann, 2010.<br>3. H. Sharp, Y. Rogers, J. Preece, Interaction Design: Beyond Human - Computer Interaction, 3rd edition, Wiley, 2011.<br>4. D. Norman, The Design of Everyday Things, Currency/ Doubleday, 1990.</td></tr>
</table>