Fifth Semester B.E. – Scheme

Sl. No.	Course Code	Course	Teaching Dept.	L-T-P-S (Hrs/week)	Total Credits	Marks
1	15IST51	Computer Networks	IS	3-0-0-0	3	100
2	15IST52	Microcontrollers	IS	3-0-0-0	3	100
3	15ISI53	Operating System (IC)	IS	3-0-2-0	4	100
4	15IST54	SoftwareEngineeringandTesting	IS	3-0-0-0	3	100
5	15ISI55X	Foundation Elective-IV(IC)	IS	3-0-2-0	4	100
6	15IST56X	Engineering Elective-V	IS	3-0-0-0	3	100
7	15ISL57	Computer Networks Laboratory	IS	1-0-2-0	2	100
8	15ISL58	Microcontroller Laboratory	IS	1-0-2-0	2	100
9	15ISH59	General Aptitude	IS/BS&H	2-0-0-0	2	100
TOTAL				**22-0-8-0**	**26**	**900**

Foundation Elective-IV (IC)

Sl. No.	Course Code	Course
1	15ISI551	Advanced Algorithms
2	15ISI552	Object Oriented Programming with JAVA
3	15ISI553	Compiler Design(NPTEL/MOOCS)

Engineering Elective-V / PBL

Sl. No.	Course Code	Course
1	15IST561	Operations Research
2	15IST562	Object Oriented Modeling and Design
3	15IST563	Computer Architecture (MOOCS)/ Information Security (MOOCS)

Seventh Semester B.E. – Scheme

Sl. No.	Course Code	Course	Teaching Dept.	L-T-P-S (Hrs/week)	Total Credits	Marks
1	15IST71	Internet of Things	IS	3-0-0-0	3	100
2	15IST72	Image Processing	IS	3-0-0-0	3	100
3	15ISI73X	Foundation Elective-IX (IC)	IS	3-0-2-0	4	100
4	15IST74X	Engineering Elective-X	IS	3-0-0-0	3	100
5	15HOE75X	Open Elective-XI	IS/BS&H/ME	2-0-0-4	3	100
6	15HOE76X	Open Elective-XII	IS/BS&H	2-0-0-4	3	100
7	15ISL77	Internet of Things Laboratory	IS	1-0-2-0	2	100
8	15ISL78	Image processing Laboratory	IS	1-0-2-0	2	100
9	15ISP79	Project Phase-I and Seminar	IS	0-0-6-0	3	100
		Total		**18-0-12-8**	**26**	**900**

Foundation Elective-IX (IC)

Sl. No.	Course Code	Course
1	15ISI731	Soft Computing
2	15ISI732	Big Data
3	15ISI733	Web Technologies – Servlet, JSP

Engineering Elective-X / PBL

Sl. No.	Course Code	Course
1	15IST741	System Modeling and Simulation
2	15IST742	Machine Learning (NPTEL/MOOCS)
4	15IST743	Project Planning and Control (MOOCS)

Open Elective-XI

Sl. No.	Course Code	Course
1	15HOE751	Tax Management
2	15HOE752	Assessment of Building Energy Performance (Offered by ASHRAE)
3	15HOE753	National Disaster Management and Mitigation
4	15HOE754	Online certification courses from IITs / IISc / SWAYAM / EDX

Open Elective-XII

Sl. No.	Course Code	Course
1	15HOE761	Small and Medium Enterprise Management
2	15HOE762	Occupational Safety and Health Administration
3	15HOE763	Animation and Multimedia Engineering
4	15HOE764	Online certification courses from IITs / IISc / SWAYAM / EDX

Fifth Semester B.E. – Syllabus

Computer Networks

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST51	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand the basics concept of data communications.
- Understand OSI and TCP/IP models.
- Understand the functions of data link layer and network layer.
- Become familiar with the basics of packet switching.
- Understand the concepts of network security and network management and its applications.

Syllabus

Module – I

Introduction and Physical Layer: Data Communications, Networks, Layered Tasks, The OSI model, Layers in OSI model, TCP/IP Protocol suite, Addressing, Transmission Impairment, Data Rate limits, Performance, Digital-digital conversion (Only Line coding: Polar, Bipolar and Manchester coding) **09 Hours**

Module – II

Switching and Data Link Layer-1: Introduction to switching, Datagram Networks, Virtual Circuit Networks, and Error Detection andCorrection: Introduction, Block coding, Cyclic codes, Checksum. **09 Hours**

Module – III

Data Link Layer-2: Framing, Flow and Error Control, Protocols, Noiseless Channels, Noisy channels, HDLC, PPP (Framing, Transition phases only) Random access, Controlled Access, Channelization,Connecting devices. **08 Hours**

Module – IV

Network Layer: Logical addressing, IPv4 addresses, IPv6 addresses, IPv4 and IPv6 Headers, Routing in Packet networks, Shortest path routing: Bellman-Ford algorithm, Dijikstra's Algorithm, Overview of network security, secret key encryption protocol, public key encryption protocols. **08 Hours**

Module – V

Applications Layer and Network Management: Application layer overview, Domain Name System (DNS), Remote Login Protocols, E-mail, FTP, World Wide Web and HTTP, Network management.Overview of Wireless Ad-Hoc networks, Routing in AdHoc Networks, Routing protocols for AdHoc networks **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Solve problems on shortest path routing algorithms.
- Analyse traffic management at different levels.
- Explain concepts of Cryptography algorithms.
- Analyse the various application layer protocols.
- Analyse different networks.

Text Books:

1. Behrouz A. Forouzan: "Data Communication and Networking", 5th Edition, Tata McGraw-Hill, July 2012, ISBN: 978-0-07-337622-6, (Chapters 1-4,8,11-13, Listed topics only).
2. Alberto Leon-Garcia, Indra Widjaja: "Communication Networks -Fundamental Concepts and Key architectures", 2nd Edition, Tata McGraw-Hill, 2004, reprint 207, ISBN-13: 978-0-07-059501-9, (Chapters 7, Listed topics only).
3. Nader F. Mir: "Computer and Communication Networks", 2nd Edition, Pearson Education, 2015, ISBN: 0133814742, (Chapters 9,10,19, Listed topics only).

Reference Books:

1. William Stallings: "Data and Computer Communication", 8th Edition, Pearson Education, 207, ISBN-13: 978-0133506488.
2. Larry L. Peterson and Bruce S. Davie: "Computer Networks – A Systems Approach", 4th Edition, Elsevier, 207, ISBN: 978-0-12-385059-1.

E-Resources:

1. https://archive.org/details/Data.Communications.and.Networking.5th.Edition
2. https://doc.lagout.org/network/Data%20Communications%20and%20Networking%20By%20Behrouz%20A.Forouzan.pdf
3. http://ptgmedia.pearsoncmg.com/images/9780133814743/samplepages/9780133814743.pdf

Microcontrollers

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST52	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand controlled operations of embedded systems using processors/ microcontrollers.
- Understand the architecture of a microcontroller 8051.
- Understand the programming model and instruction set supported by the microcontroller.

Syllabus

Module – I

8085 Microprocessor architecture: 8085 Microprocessor Detailed functional block Diagram and Architecture, the 8085 Programming Model, Instruction, opcode and data format.

Microprocessors and Microcontroller: RISC and CISC CPU architectures, Harvard and Von Neuman CPU Architecture. **08 Hours**

Module – II

The 8051 architecture: Introduction, 8051 Microcontroller hardware, I/O pins, ports and circuits, external memory

8051 Addressing modes and operations: Introduction, Addressing modes, External Data moves, Code memory, read only data moves, Indexed addressing mode.

08 Hours

Module – III

PUSH and POP opcodes, Data exchanges, example programs. Logical operations introduction, Byte level, bit level logic operations, Rotate and Swap operation ,example programs, Arithmetic operation introduction, flags, incrementing and decrementing, addition subtraction, multiplication and division, Decimal arithmetic, Example programs. **08 Hours**

Module – IV

Jump and call operations: The JUMP and CALL program range, Jump calls and subroutine and returns, example programming

8051 programming in C: Data types and time delays in 8051C,I/O programming, Logic operations, Data conversion programs, Accessing code ROM and Data serialization

Timer/Counter programming in 8051: Programming 8051 timers Counter programming, Programming timer 0 and 1. **08 Hours**

Module – V

8051 Serial communication: Basics of serial communication, 8051 connections to Rs-232, 8051 serial communication programming.

Interrupt programming: 8051 interrupts, Programming Timer Interrupts, programming External hardware Interrupts, programming the serial communication interrupt, interrupt priority in 8051. **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Explain the architecture of microprocessor and its comparison with microcontroller.
- Explain complete architecture of microcontroller.
- Write assembly level programs for 8051 microcontroller.
- Develop a C code for microcontroller based system.
- Formulate interrupt programs with 8051 hardware and software interrupts.

Text Books:

1. Ramesh S Gaonkar: "Microprocessor Architecture, Programming and Applications with 8085", 6th Edition, (Chapter 2), Prentice Hall, 2013.
2. Kenneth J Ayala: "The 8051 Micro controller Architecture, Programming and application", 3rd Edition, (Chapters 3.1-3.3,5.1-5.5,6.1-6.3,7.1-7.6), Thomson Learning.
3. Muhammad Ali Mazidi Janice Gillispie Mazidi, Roolin D McKinlay: "The 8051 Micro Controller And embedded Systems - using assembly and C", 2nd Edition, (Chapters 9-11), Pearson Education, 2007.

Reference Book:

1. Michael Predko: "Programming and Customizing the 8051 Microcontroller", Mc-Graw Hill, 1999.

E-Resources:

1. https://en.wikibooks.org/Embedded-systems/8051-microcontroller
2. www.circuitstoday.com/ 8051-microcontroller
3. nptel.ac.in/courses/webcourse-contents/IIT-KANPUR/microcontrollers

Operating System (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI53	3-0-2-0	4	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Learn various concepts of operating system.
- Learn the concepts of Process synchronization.
- Gain knowledge about deadlock's occurring in resource allocation.
- Understand the memory management function of operating system.
- Realize the role of operating system in file management.

Syllabus

Module – I

Introduction to Operating Systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments. Operating System Services; Operating System structure; Virtual machines; Operating System generation; System boot. **08 Hours**

Module – II

Process Management: Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling. **08 Hours**

Module – III

Process Synchronization: Synchronization: The critical section problem, semaphores, classical problems of synchronization.

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **09 Hours**

Module – IV

Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Page replacement; Allocation of frames. **08 Hours**

Module – V

File System, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. **07 Hours**

Laboratory

Note: The following programs can be executed on C/C++ any equivalent language or tool with suitable platform.

1. Write a program to simulate the Round Robin Scheduling algorithm.
2. Write a program to implement the Shortest Job First algorithm.
3. Write a program to implement the First come First Serve algorithm.
4. Write a program to implement the Priority Scheduling algorithm.
5. Design Develop and run a program to implement Banker's algorithm.
6. Write a Program to implement Inter process Communication using Pipes.
7. Write a Program to implement Producer Consumer Problem.
8. Installation of Operating systems.

Course Outcomes:

On completion of this course, students will be able to :

1. Explain the concepts of Operating System Structure, Operations and Services.
2. Design new techniques for Multithreaded Programming, Process Scheduling and Synchronization.
3. Apply the skills of deal lock prevention and avoidance.

4. Design and implement Memory management Algorithms.
5. Explain the concept of file systems.

Text Book:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: "Operating System Principles", Wiley India, (Chapters 1-12), 8th Edition, 2009, ISBN: 9781118063330.

Reference Books:

1. D.M Dhamdhere: "Operating systems - A concept based Approach", Tata McGraw- Hill, 2nd Edition, 2002. ISBN: 978–0–07–295769–3.
2. P.C.P. Bhatt: "Introduction to Operating Systems: Concepts and Practice", (Chapters: 1, 2, 3.1 to 3.4 , 4.1 to 4.4, 5.1 to 5.5, 6.1 to 6.7, 7, 8.1 to 8.6, 9.1 to 9.6, 10, 11.1 to 11.5, 12.1 to 12.6, 14.1 to 14.8, 21.1 to 21.9) , PHI, 4th Edition, 208. ISBN: 978-81-203-4836-3.
3. Harvey M Deital: "Operating systems", Pearson Education, 3rd Edition, 1990. ISBN 978-0131828278.

E-Resources:

1. http://nptel.ac.in/courses/106108101/
2. http://study.com/academy/lesson/computer-operating-systems-managing-hardware-and-software-resources.html

Software Engineering and Testing

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
17IST54	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand the real world applications with the aid of software engineering techniques along with professional ethics and responsibilities.
- Understand the importance of SDLC life cycle in realtime.
- This course will enable students to :describe Software and Hardware Testing.

Syllabus

Module – I

Introduction: FAQ's about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems. Critical Systems: A simple safety critical system; System dependability; Availability and reliability. Software processes: models, process iteration, activities. **08 Hours**

Module – II

Requirements and Project Management: Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; The software requirements document. Project Management: Management activities; Project planning; Project scheduling. **08 Hours**

Module – III

Software Development and Verification and Validation: Agile methods; Extreme programming; Rapid application development. Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. **08 Hours**

Module – IV

A Perspective on Testing, Examples: Basic definitions, Test cases, Software testing: Component testing; Test case design; Test automation, Levels of testing.

Boundary Value Testing: Boundary value analysis, Robustness testing, Worst-case testing, Special value testing. **08 Hours**

Module – V

Equivalence Class Testing, Decision Table- Based Testing: Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations.

Path Testing, Data Flow Testing: DD paths, Test coverage metrics, Basis path testing.

08 Hours

Course Outcomes:

On completion of this course, students will be able to :

- Understand the real world applications with the aid of software engineering techniques along with professional ethics and responsibilities.
- Learn the requirements engineering process, feasibility studies and its validation.
- Design software in structured, organized ways and demonstrate effective, practical ways to design and develop high quality software.
- Write Test cases, Test plan based on test scenario.
- Analyze using static analysis tools (Inspection, walk through and peer review) and automate high quality tests during unit and integration testing.

Text Books:

1. Ian Sommerville: "Software Engineering", 8th Edition, (Chapters 2,3,5,6,17,22), Pearson Education, 207, ISBN-13: 978-0137035151.
2. Paul C. Jorgensen: "Software Testing, A Craftsman's Approach", 3rd Edition, (Chapters 1,2,5,6,7,9,10), Auerbach Publications, 208, ISBN: 9781439889510.

Reference Books:

1 Roger.S.Pressman: "Software Engineering - A Practitioners approach", 7th Edition, McGraw Hill, 207.
2. Pankaj Jalote: "An Integrated Approach to Software Engineering", Wiley India, 2009.
3. Myers GJ: "The Art of Software testing", Wiley-Dreantech India Pvt. Ltd., 2004.
4. LoiseTamres: "Introducing Software Testing", Pearson Education, 2003.

E-Resources:

1. nptel.ac.in/courses/106101061
2. computingcareers.acm.org
3. www.tutorialspoint.com/software-engineering

Advanced Algorithms (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI551	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Learn methods for solving recurrences, which are useful for describing the running times of recursive algorithms.
- Understand the graph search algorithms and network flow problems.
- Gain knowledge about number - theoretic algorithms
- Get exposed to string - matching algorithms.
- Introduce probabilistic analysis and randomized algorithms..

Syllabus

Module – I

Introduction: The role of algorithms in computing, Growth of Functions: Asymptotic notations; Standard notations and common functions; Methods for solving recurrences: The substitution method, The recurrence – tree method, The master method.

Amortized Analysis: Aggregate, Accounting and Potential Methods. **09 Hours**

Module – II

Graph Algorithms: Introduction to Single – Source shortest paths: Variants, negative weight edges, Cycles, Representing Shortest paths, Relaxation, Bellman - Ford Algorithm; Single source shortest paths in a DAG; Dijkstra's algorithm, Johnson's algorithm.

Maximum Flow: Flow networks and Ford-Fulkerson method; Maximum bipartite matching. **08 Hours**

Module – III

Number-Theoretic Algorithms: Elementary notions; GCD; Modular Arithmetic; Solving modular linear equations; The Chinese remainder theorem; Powers of an element; RSA cryptosystem; Primality testing. **08 Hours**

Module-IV

Number-Theoretic Algorithms: Elementary notions; GCD; Modular Arithmetic; Solving modular linear equations; The Chinese remainder theorem; Powers of an element; RSA cryptosystem; Primality testing. **08 Hours**

Module – V

String-Matching Algorithms: Naïve string Matching; Rabin - Karp algorithm; String matching with finite automata; Knuth-Morris-Pratt algorithm; Boyer – Moore algorithms. **07 Hours**

Laboratory

1. Design, develop, and run a program in any language to implement the Bellman-Ford algorithm.
2. Design, develop, and run a program in any language to implement a Monte Carlo algorithm to test the primality of a given integer.
3. Design, develop, and run a program in any language to solve modular linear equations.
4. Design, develop, and run a program in any language to solve string matching problem using naïve approach and the KMP algorithm.
5. Design, develop, and run a program in any language to solve String matching problem using Finite Automata.
6. Design, develop, and run a program in any language to solve String matching problem using Robin Karp algorithm.

Course Outcomes:

On completion of this course, students will be able to :

- Explain different asymptotic notations and their use in modern computing systems.
- Design and apply iterative and recursive algorithms.
- Design and implement graph and flow network algorithms.
- Design and analyze the algorithms for string matching.
- Describe the representation of polynomials, the DFT and alsothe implementation of FFT.

Text Books:

1. T. H Cormen, C E Leiserson, R L Rivest, C Stein: "Introduction to Algorithms", (Chapters 1,3,4,17,24-26,30-32), Prentice-Hall of India, 3rd Edition, 2010, ISBN: 9780262259460.
2 Kenneth A. Berman, Jerome L. Paul: "Algorithms", (Chapters 1,11,20), Cengage Delmar Learning, India, 2002, ISBN: 9788131505212.

Reference Book:

1. Ellis Horowitz, Sartaj Sahni, S.Rajasekharan: "Fundamentals of Computer Algorithms", 2nd Edition, Universities press, 207, ISBN: 9788173716126.

E-Resources:

1. http://staff.ustc.edu.cn/~csli/graduate/algorithms/book6/chap02.htm
2. http://www.cs.cornell.edu/courses/cs3110/2011sp/lectures/lec20-amortized/amortized.htm
3. http://staff.ustc.edu.cn/~csli/graduate/algorithms/book6/chap18.htm
4. https://en.wikipedia.org/wiki/Category:Graph_algorithms
5. http://staff.ustc.edu.cn/~csli/graduate/algorithms/book6/chap18.htm
6. http://staff.ustc.edu.cn/~csli/graduate/algorithms/book6/chap33.htm
7. https://en.wikipedia.org/wiki/String_searching_algorithm

Object Oriented Programming with JAVA (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI552	3-0-2-0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Understand the basic concepts of Java Technology and its features.
- Get a clear understanding of the OOPs concepts.
- Write programs in Java.
- Effectively use data structures like Collections, Lists, etc.
- Write defensive programming using Exception Handling.

Syllabus

Module – I

Introduction to Java: Why Java, Flavors of Java, Java Designing Goal and Features, JVM / JDK / JRE / History of JDK / JDM, Usage of IDE (Eclipse, NetBeans)

Language Fundamentals: Data Types - Variables, keywords, operators; Selection / Iterative / Decision making statements

Introduction to OOPs Concepts: Inheritance - Polymorphism - Abstraction – Encapsulation **12 Hours**

Module – II

Arrays and Strings: Defining of an Array, Initializing and accessing an Array, Multi-Dimensional Array, String / String Buffer / String Builder.

OOPs in Java: Inheritance, Abstract class and interface, Abstract class Vs Interface.

Packages and Wrapper Classes: Defining Package, Organizing Classes and interfaces in Packages, Package as Access Protection, Import and Static Import, Naming Convention for packages, What is Wrapper Class, Why Wrapper, How to handle wrapper Classes. **11 Hours**

Module – III

Exception Handling: What is Exception, Types of Exception, Exception Hierarchy, Custom excpetions.

The Collection Framework: Collection of objects, Collection Interfaces and Hierarchy, List and Map, Types of List, Types of Map, Iterator, Generics. **08 Hours**

Module – IV

Threads: Understanding Threads, Needs of Multi-Threaded Programming, Thread Life-cycle, Synchronizing Threads. **03 Hours**

Module – V

Project Work: To create a Contact Book application using the Core Java concepts learnt with special emphasis on OOPs concepts, Exception Handling, and Collections Framework. **24 Hours**

Laboratory

1. Programs covering Data Types and OOPs Concepts.
2. Programs covering Arrays and Strings, OOPS concepts in Java, Packages and Wrapper Classes.
3. Programs covering Exception Handling, The Collection Framework and Threads.

Course Outcomes:

On completion of this course, students will be able to :

- Understand the basic concepts of Java Technology and its features.
- Get a clear understanding of the OOPs concepts.
- Write programs in Java.
- Effectively use data structures like Collections, Lists, etc.
- Write defensive programming using Exception Handling.

Text Books:

1. Herbert Schildt: "JAVA: The Complete Reference", McGraw Hill Education, 9th Edition, ISBN-10: 9339212096.
2. Dr. R. Nageswara Rao: "Core Java: An Integrated Approach", Dreamtech Press, 1st Edition, 2016, ISBN-10: 9351199258.

Reference Books:

1. Joshua Bloch: "Effective Java", Pearson Education, 2nd Edition, ISBN-10: 933257653X
2. Cay S. Horstmann: "Core Java - Vol. I - Fundamentals", Pearson Education, 10th Edition, ISBN-10: 9332582718.

Operations Research

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST561	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Understand quantitative methods and techniques for effective decision- making, model formulation and applications that are used in solving real world problems.
- Know the various techniques of OR,their applications and the relationship between a linear program and its dual.
- Learn different types of transportation and assignment models for optimization.
- Understand techniques that are used to plan, schedule and monitor large projects such as building construction, maintenance of computer system, research and development design etc.
- Acquire knowledge on decision making techniques under conflicting situations where there are one or more opponents.

Syllabus

Module – I

Introduction, Linear Programming: Introduction: The Origins , Nature and Impact of OR; Defining the Problem and Gathering Data; Formulating a Mathematical Model; Deriving Solutions from the Model; Testing the Model; Preparing to Apply the Model; Implementation. Linear Programming: Prototype Example; The Linear Programming Model; Assumptions of Linear Programming; Additional Examples. **08 Hours**

Module – II

The Simplex Method: The Essence of the Simplex Method; Setting Up the Simplex Method; The Algebra of the Simplex Method; The Simplex Method in Tabular Form; Tie Breaking in the Simplex Method; Adapting to Other Model Forms;. Duality Theory: The Essence of Duality Theory; Primal-Dual Relationships; Adapting to Other Primal Forms; The Dual Simplex Method. **08 Hours**

Module – III

Transportation and Assignment Problems: The Transportation Problem; The Assignment Problem. **08 Hours**

Module – IV

Project Management with PERT/CPM: A Prototype Example--- The Reliable Construction Co. Project; Using a Network to Visually Display a Project; Scheduling a Project with PERT/CPM; Dealing with Uncertain Activity Durations; Considering Time-Cost Trade-Offs; Scheduling and Controlling Project Costs; An Evaluation of PERT/CPM. **08 Hours**

Module –V

Game Theory, Decision Analysis: Game Theory: The Formulation of Two-Person, Zero-Sum Games; Solving Simple Games--A Prototype Example; Games with Mixed Strategies; Graphical Solution Procedure; Solving by Linear Programming; Extensions. Decision Analysis: A Prototype Example; Decision Making without Experimentation; Decision Making with Experimentation; Decision Trees. **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Develop Linear Programming models, interpret the models, solutions and infer solutions to the real-world problems.
- Solve the Linear problems by applying different techniques of Operations research.
- Build and solve Transportation models and Assignment models.
- Design new simple models like CPM to improve decision making and use critical path analysis, programming evaluation and review techniques for timely project scheduling and completion.
- Compare the characteristics of different types of decision making environments and the appropriate decision making approaches and tools to be used in each type

Text Book:

1. Frederick S. Hillier and Gerald J. Lieberman: "Introduction to Operations Research: Concepts and Cases", 8th Edition, Tata McGraw Hill, 2005, ISBN-13: 978-0-07-060092-8, (Chapters 1.1 to 1.3, 2, 3.1 to 3.4, 4.1 to 4.6, 6.1 to 6.4, 7.1, 8,10, 14.1 to 14.6, 15.1 to 15.4).

Reference Books:

1. S D Sharma: "Operations Research", KedarNath RamNath, 207.
2. Hamdy A Taha: "Operations Research: An Introduction", 8th Edition, Pearson Education, 2007, ISBN: 81-203-2235-5.
3. Richard Bronson, Govindasami Naadimuthu: "Operations Research", SCHAUM'S Outlines, 2nd Edition, Tata Mcgraw-Hill, ISBN-13:978-0-07-058400-6, ISBN-10: 0-07-058400-1.

Object Oriented Modeling and Design

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST562	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Introducing students to the concepts of object oriented approach to perform systems analysis and design
- Highlighting the importance and limitations of object-oriented analysis and design.
- Show how object oriented analysis and design is applied in development of software.
- Pointing out the importance of UML model through out the process of object oriented modeling and design.
- Providing students with the necessary knowledge and skills in using object-oriented CASE tools

Syllabus

Module-I

Introduction, Modeling Concepts, class Modeling: What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history

Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models. **07 Hours**

Module-II

Advanced Class Modeling, State Modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages.

State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; examples. **08 Hours**

Module-III

Advanced State Modeling, Interaction Modeling: Advanced State.

Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models.

Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models.

08 Hours

Module-IV

Process Overview, System Conception, Application Analysis: Process Overview: Development stages; Development life cycle.

System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement.

Application Analysis: Application interaction model, Application class model, application state model, adding opertoins. **08 Hours**

Module-V

System Design: Application Design and implementation modeling : Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing. **08 Hours**

Object oriented modeling and design Lab

To develop a mini-project following the 12 exercises listed below.

1. To develop a problem statement.
2. Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).
3. Identify Use Cases and develop the Use Case model.
4. Identify the business activities and develop an UML Activity diagram.
5. Identity the conceptual classes and develop a domain model with UML Class diagram.
6. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.
7. Draw the State Chart diagram.
8. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.
9. Implement the Technical services layer.
10. Implement the Domain objects layer.
11. Implement the User Interface layer.
12. Draw Component and Deployment diagrams.

Suggested domains for Mini-project.

1. Passport automation system.
2. Book bank
3. Exam Registration
4. Stock maintenance system.
5. Online course reservation system
6. E-ticketing
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference Management System

Course Outcomes:

On completion of this course, students will be able to :

- Construct models to Show the importance of systems analysis and design in solving complex problems
- Recognize the difference between various object relationships: inheritance, association, and dependency relationships.
- Represent an object-oriented system using number of modeling views.
- Analyze the role and function of each UML model in developing object-oriented software.
- Estimate System performance and construction of UML models and expressing the appropriate notation associated with each model.

Text Book:

1. Michael Blaha, James Rumbaugh: "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education, 2005, ISBN 10: 8131711064 ISBN 13: 97 88131711064

Reference Books:

1. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: :UML 2 Toolkit", Wiley- Dreamtech India, 2004, ISBN-10: 0471463612, ISBN-13: 9780471463610.
2. Grady Booch: "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007, ISBN-10: 8131722872, ISBN-13: 9788131722879.

E- Resources:

1. http://www.tutorialspoint.com/object_oriented_analysis_design
2. https://www.slideshare.net/helghareeb/object-oriented-analysis-and-design
3. https://www.lynda.com/Java-tutorials/Understanding-object-oriented-analysis-design-processes/96949/106067-4.html

Computer Networks Laboratory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISL57	1-0-2-0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Become familiar with the working of network topology.
- Understand working of Ethernet LAN.
- Understand the concepts of mobile routing.
- Become familiar with the basics of TCP/IP.
- Understand the concepts of network security.

Part A

Simulation Exercises

The following experiments shall be conducted using either NS228/OPNET or any other suitable simulator.

1. Simulate a three nodes point – to – point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate a four node point-to-point network with the links connected as follows: n0 – n2, n1 – n2 and n2 – n3. Apply TCP agent between n0-n3 and UDP between n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets sent by TCP / UDP.
3. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
4. Simulate an Ethernet LAN using n nodes (6-10), change error rate and data rate and compare throughput.
5. Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
6. Simulate simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.

Part B

Implement the following in C/C++:

1. Write a program for error detecting code using CRC-CCITT (16- bits).
2. Write a program for distance vector algorithm to find suitable path for transmission.

3. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.
4. Implement the above program using as message queues or FIFOs as IPC channels.
5. Write a program for simple RSA algorithm to encrypt and decrypt the data.
6. Write a program for congestion control using leaky bucket algorithm.

Course Outcomes:

On completion of this course, students will be able to :

- Analyse the working of network devices.
- Differentiate packet managements at different levels.
- Apply the knowledge of security algorithms.
- Discover shortest path using routing algorithms.
- Demonstrate the working of wireless networks.

Text Books:

1. Behrouz A. Forouzan: "Data Communication and Networking", 5th Edition, Tata McGraw-Hill, 2012, ISBN: 978-0-07-337622-6.
2. Alberto Leon-Garcia, Indra Widjaja: "Communication Networks -Fundamental Concepts and Key architectures", 2nd Edition,, Tata McGraw-Hill, 2004, Reprint 2007, ISBN-13: 978-0-07-059501-9.
3. Nader F. Mir: "Computer and Communication Networks", 2nd Edition, Pearson Education, 2015, ISBN: 0133814742.

E- Resources:

1. https://archive.org/details/Data.Communications.and.Networking.5th.Edition
2. https://doc.lagout.org/network/Data%20Communications%20and%20Networking%20By%20Behrouz%20A.Forouzan.pdf
3. http://ptgmedia.pearsoncmg.com/images/9780133814743/samplepages/9780133814743.pdf

Microcontroller Laboratory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISL58	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Hands-on experience on the theoretical concepts through simple experiments

List of Experiments:

1. Perform the following,
   a. Add two multibyte number.
   b. Subtract two multibyte numbers.
2. Operations on the data block,
   a. Data block movement.
   b. Data block Exchange.
3. Perform the following,
   a. To find square of a given number.
   b. To find number of 1's and 0's in a given data.
4. Data conversion:
   Convert binary(hex) number to decimal equivalent.
5. To find the largest /smallest in an array,
6. To arrange the given array in ascending / descending order.
7. Counters:
   BCD up and down counter.
8. Write a program to show serial data transmission using 8051 microcontroller.
9. Interface DAC to 8051 chip to generate,
   a. Triangular waveform.
   b. Square wave

Course Outcomes:

On completion of this course, students will be able to :

- Demonstrate simple arithmetic and logical operations using 8051 MC.
- Generate waveforms using 8051 MC.
- Perform number conversion and counters using 8051 MC.
- Interface 8051 MC with motors.

Text Books:

1. Ramesh S Gaonkar: "Microprocessor Architecture, Programming and Applications with 8085", 6th Edition, Prentice Hall, 2013.
2. Kenneth J Ayala: "The 8051 Micro controller Architecture, Programming and application", 3rd Edition, Thomson Learning.
3. Muhammad Ali Mazidi Janice Gillispie Mazidi, Roolin D McKinlay: "The 8051 Micro Controller And embedded Systems - using assembly and C", 2nd Edition, Pearson Education, 2007.

Reference Book:

1. Michael Predko: "Programming and Customizing the 8051 Microcontroller", McGraw Hill, 1999.

E-Resources:

1. https://en.wikibooks.org/Embedded-systems/8051-microcontroller
2. www.circuitstoday.com/ 8051-microcontroller
3. nptel.ac.in/courses/webcourse-contents/IIT-KANPUR/microcontrollers

General Aptitude

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISH59	2:0:0:0	2	CIE:50 SEE:50	3 Hours	HSS

Course Objectives:

This course will enable students to :

- Understand different types of Numerical / Arithmetical problems.
- Understand the different Data interpretation problems.

Syllabus

Module – I

Numerical Ability-I: Numbers, HCF and LCM of numbers, Decimal Fractions, Average, Problems on Numbers, Problems on Ages. **06 Hours**

Module – II

Numerical Ability-II: Percentage, Profit and Loss, Ratio and Proportion, Partnership, Chain Rule, Time and Work. **05 Hours**

Module – III

Numerical Ability-III: Pipes and Cistern, Time and Distance, Problems on Trains, Alligation or Mixture, Simple Interest, Compound Interest. **05 Hours**

Module – IV

Numerical Ability-IV: Races and Games of Skill, Calender, Clocks, Permutations and Combinations, Probability, Odd man out and Series. **05 Hours**

Module-V

Data Interpretation: Tabulation, Bar Graphs, Pie Charts, Line Graphs. **05 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Solve and analyze different types of Numerical / Arithmetical problems.
- Solve and analyze different Data interpretation problems.

Text Books:

1. R S Aggarwal: "Quantitative Aptitude for competitive examinations", (Chapters 1-3,6-8,10-18,20-22,26-28,30,31,35-39), S. Chand Publishing, New Delhi, 2014, ISBN-13: 978-81-219-2498-6.

ଓ ଓ ଓ ଓ ଓ

Sixth Semester B.E. – Syllabus

Unix System Programming

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST61	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Understand UNIX, ANSI and POSIX standards.
- Write various programs to manipulate files and their attributes.
- Understand process environment, process control and process relationship.
- Understand the concept of signal handling and daemon process and its characteristics.
- Analyze interposes communication concepts pipes, message queues, shared memory, semaphores.

Syllabus

Module – I

Introduction: UNIX and ANSI Standards: The ANSI C Standard, the ANSI/ISO C++ Standards, Difference between ANSI C and C++, the POSIX Standards, the POSIX.1 FIPS Standard, the X/Open Standards. UNIX and POSIX APIs: The POSIX APIs, API Common Characteristics, File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files. **08 Hours**

Module – II

UNIX files and UNIX file API's: UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links. General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. **08 Hours**

Module – III

UNIX Process and Control: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes. Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs. **08 Hours**

Module – IV

Process relationship and Signals: Interpreter Files, system Function, Process Accounting, User identification, Process Times, Terminal Logins, Network Logins, Process Groups, Sessions, controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, Kill, Alarm. **08 Hours**

Module – V

Inter-Process Communication: Introduction, Pipes, popen, and pclose Functions, Coprocesses, FIFOs, Message Queues, and Semaphores, shared memory. Daemon Characteristics, Coding Rules, basics of socket programming. **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Analyze the different manifested constants to determine the system is POSIX conforming.
- Differentiate the ANCSI C and POSIX standard.
- Use API's to implement interposes communication and other basic services of UNIX kernel.
- Implement fork, race condition, zombie process etc.
- Apply system calls to create processes that manipulate system resources and get the system configuration limits.

Text Books:

1. Terrence Chan: "UNIX System Programming Using C++", Prentice Hall India, 1999, ISBN-10: 0133315622, (Listed topics only from Chapters 1,5-10).
2. W. Richard Stevens: 'Advanced Programming in the UNIX Environment", 2nd Edition, Pearson Education, 2005,ISBN: 0201433079.(Listed topics only from Chapters 7, 8, 9, 13, 14,15)

Reference Books:

1. Marc J. Rochkind: "Advanced UNIX Programming", 2nd Edition, Pearson Education, 2005.
2. Maurice J Bach: "The Design of the UNIX Operating System", Pearson Education, 1987.

E-Resources:

1. www.compsci.hunter.cuny.edu/~sweiss/course_materials/unix_lecture.../chapter_01.pdf
2. www.cs.northwestern.edu/~pdinda/netclass-f00/unix_nutshell.pdf
3. www.goodreads.com/author/show/1268629.Terrence_Chan
4. www.adamsenggcollege.ac.in/Unix%20System%20Programming%20NOTES%20b

Android Programming (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI62	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Describe the Android SDK features and the Development Framework and understanding Activities.
- Create adaptive, responsive user interfaces that work across a wide range of devices.
- Perform background work and long-running tasks in Android applications
- Know the concepts of Storing, sharing and retrieving data in Android applications
- Learn how permissions, security and performance affect application. Finally, make sure your app is ready to share with the world, and publish it.

Syllabus

Module- I

What Is Android?, Android Versions, Features of Android, Architecture of Android, Android Devices in the Market, The Android Market, Obtaining the Required Tools, Eclipse, Android SDK, Android Development Tools (ADT), Creating Android Virtual Devices (AVDs), Creating Your First Android Application, Anatomy of an Android Application. Understanding Activities, Applying Styles and Themes to Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Linking Activities Using Intents, Resolving Intent Filter Collision, Returning Results from an Intent**.** **08 Hours**

Module- II

Understanding the Components of a Screen, Views and View Groups, Linear Layout, Absolute Layout, Table Layout, Relative Layout, Frame Layout, Scroll View, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Creating the User Interface Programmatically, Basic Views . **08 Hours**

Module- III

Using Image Views to Display Pictures - Gallery and Image View Views, Image Switcher, Grid View, Using Menus with Views - Creating the Helper Methods, Options Menu, Context Menu, Saving and Loading User Preferences - Using get Shared

Preferences(), Using get Preferences(), Persisting Data to Files - Saving to Internal Storage, Saving to External Storage (SD Card), Choosing the Best Storage Option, Using Static Resources, Creating and Using Databases. **08 Hours**

Module- IV

Sharing Data in Android, Using a Content Provider - Predefined Query String Constants, Projections, Filtering, Sorting, Creating Your Own Content Providers - Using the Content Provider. SMS Messaging - Sending SMS Messages Programmatically, Getting Feedback After Sending the Message, Sending SMS Messages Using Intent, Receiving SMS Messages, Updating an Activity from a Broadcast Receiver, Invoking an Activity from a Broadcast Receiver. **09 Hours**

Module- V

Creating Your Own Services - Performing Long-Running Tasks in a Service, Performing Repeated Tasks in a Service, Executing Asynchronous Tasks on, Separate Threads Using Intent Service, Communicating between a Service and an Activity, Binding Activities to Services. Preparing for Publishing, Versioning, Digitally Signing Your Android Applications, Deploying APK Files - Using the adb.exe Tool, Using a Web Server, Publishing on the Android Market, Creating a Developer Profile, Submitting Your Apps. **07 Hours**

List of Experiments

Programs supplement the lecture concepts will be based on the latest version of Android SDK.

1. Install Android Studio and Run Hello World
2. Create and Start Activity Lifecycle and Instance State
3. Create Implicit Intents
4. Make Your First Interactive UI Using Layouts and Text View Elements
5. Using An Options Menu
6. Create a Recycler View
7. Drawables, Themes and Styles
8. Create an AsyncTask
9. Connect to the Internet
10. Broadcast Receiver
11. Set and retrieve shared preferences
12. Implement a simple content provider

Course Outcomes:

On completion of this course, students will be able to :

- Comprehend the basic features of Android Platform and Create Activities in Android.
- Demonstrate the design concepts of user interface using components and views in Android.
- Create and use databases for Android Application.
- Implement messaging services in Android.
- Deploy mobile applications in various marketplace for distribution.

Text Books:

1. Wei – Meng Lee: "Beginning Android Application Development", Wiley publications, ISBN: 978-1-118-01711-1, (Chapters 1-8,10,11).
2. Reto Meier: "Professional Android 4 Application Development", Wiley publications Publisher, 2012, ISBN-10: 812653608X.

Reference Books:

1. Mark Murphy: "Beginning Android 3", Apress Springer India Pvt. Ltd., 1st Edition, 2011, ISBN-13: 978-1-4302-3297-1
2. Sayed Hashimi, Satya Komatineni, Dave MacLean; Pro Android 4; Apress Springer India Pvt Ltd; 1st Edition; 2012; ISBN: 978-1-4302-3930-7.
3. Reto Meier: "Professional Android 2 Application Development", Wiley India Pvt. Ltd., 1st Edition, 2012, ISBN: 9788126525898.
4. James Steele: "The Android Developer's Cookbook: Building Applications with the Android SDK", Addison-Wesley Professional, 2010.

E-Resources:

1. https://developers.google.com/training/adf
2. https://goo.gl/ADKvq8
3. https://innovator.samsungmobile.com

Embedded Systems

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST63	3:0:0:0	3	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Obtain a broad understanding of the embedded system technologies and applications of embedded systems
- Understand the communication buses for device networks of embedded systems.
- Have a basic knowledge on the various issues involved in real-time operating systems.
- Know how these systems can handle the tasks and scheduling of the tasks
- Learn about embedded systems associated design and software development tools.

Syllabus

Module – I

Introduction to Embedded Systems: Embedded systems; Processor embedded into a system; Embedded hardware units and devices in a system; Embedded software in a system; Examples of embedded systems; Embedded System-on-Chip (SoC), Formalization of system design; Design process and design examples; Classification of embedded systems; Skills required for an embedded system designer. **08 Hours**

Module – II

Communication Buses for Device Networks: Wireless devices; Timer and counting devices; Watchdog timer; Real time clock; Networked embedded systems; Serial bus communication protocols; Parallel bus device protocols; Internet enabled systems; Wireless and mobile system protocols. **08 Hours**

Module – III

Real Time Operating SystemS-1: Operating System services; Process management; Timer functions; Event functions; Memory management; Device, file and I/O sub-systems management; Interrupt routines in RTOS environment and handling of interrupt source calls. **08 Hours**

Module – IV

Real Time Operating Systems-2: Real-Time Operating Systems; Basic design using an RTOS; RTOS task scheduling models, interrupt latency and response times of the tasks as performance metrics; OS security issues. **08 Hours**

Module –V

Embedded Software Development Tools: Introduction; Host and target machines; Linking and locating software; Getting embedded software in to the target system; Issues in hardware-software design and co-design; Testing on host machine; Simulators; Laboratory tools. **07 Hours**

List of Lab programs

1. Intrusion detection in TCP/IP networks using immune systems paradigm and neural network detectors.
2. Network Tapping System Based on Customized Embedded Linux: Design and Implementation.
3. Network Intrusion Detection System Based on Embedded System - Off-line and On-line NIDS Based on Embedded System: Design and Implementation.
4. Packet Features Extractor for Net5work Security Systems: Design and Implementation.
5. Draft of Design and Implementation FSK Remote Control System Using ATmiga16 Microcontroller.
6. Text Code of Tx Rx remote control ATmiga16 Microcontroller.

Course Outcomes:

On completion of this course, students will be able to :

- Describe the embedded system technologies and applications of embedded systems
- Differentiate communication buses for device networks of embedded systems
- Solve issues involved in real-time operating systems.
- Manage tasks scheduling of embedded systems.
- Design and software development tools.

Text Book:

1 Raj Kamal: "Embedded Systems: Architecture, Programming and Design", Tata McGraw-Hill Education, 2011, ISBN: 070667640, ISBN-13: 978070667648.

Reference Book:

1. Shibu K V: "Introduction to Embedded Systems", Tata McGraw Hill, 2009.

E-Resources:

1. https://books.google.co.in/books/about/Embedded_Systems.html?id=pWlbvW0H3IAC
2. https://www.abebooks.com/Embedded-Systems-Architecture-Programming-Design-Raj/4994588934

Distributed Computing System (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI641	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Learn Basic Concepts of DSM, Hardware DSM.
- Understand File Sharing, DFS Implementation, Replication in DFS,
- Understand the concepts of Cryptanalysis, Secure channels, Access control.
- Understand some of the security concepts in distributed computing.
- Understand the main ideas and concepts on web services.
- Study and work on a related topic of internet applications such as information hiding, system security and E-learning.
- Understand the concepts of UDDI, SOAP, JMS remote procedure calls.

Syllabus

Module – I

Distributed System Management: Introduction, Resource management, Task Assignment Approach, Load Balancing Approach, Load-Sharing Approach, Process management in a Distributed Environment, Process Migration, Threads,Fault Tolerance. **08 Hours**

Module – II

Distributed Shared Memory: Introduction, Basic Concepts of DSM, Hardware DSM, Design Issue in DSM Systems, Issue in Implementing DSM Systems, Heterogeneous and Other DSM Systems, Case Studies. **08 Hours**

Module – III

Distributed File System: Introduction to DFS, File Models, Distributed File System Design, Semantics of File Sharing, DFS Implementation, File Caching in DFS, Replication in DFS, Case studies. **Naming**: Introduction, Desirable features of a good naming system, Basic concepts, System-oriented names, Object-locating mechanisms, Issues in designing human-oriented names, Name caches, Naming and security, Case study: Domain name service. **08 Hours**

Module – IV

Security in distributed systems: Introduction, Cryptography, Secure channels, Access control, Security Management, Case studies. **08 Hours**

Module –V

Real-Time Distributed Operating Systems: Introduction, Design issues in real-time distributed systems, Realtime communication, Real-time scheduling, Case study: Real-time communication in MARS.

Emerging Trends indistributed Computing: Introduction to emerging trends, Grid Computing, SOA, Cloud computing, the future of emerging Trends. **08 Hours**

List of Experiments

Note: Use Use EJB 3.X or any equivalent tool to implement the following experiment:

1. Design and implement client server application using RMI (Remote Method Invocation) to invoke a service to calculate the income tax.
2. Design and implement EJB (Entity Java Beans) session bean business logic to calculate income tax and invoke the service using stub, i.e., client side proxy object.
3. Design and implement an EJB entity bean to persist the client submitted data into an enterprise information system.
4. Design and implement an offline database communication system using JMS (Java Message Service) to service the client request.
5. Design and implement the client code to call the Micro soft service like free s ervice from UDDI (Universal Description Discovery Protocol).
6. Design and implement business logic and bind it as service using SOAP (Simple Object Access Protocol), also implement client to call service.

Course Outcomes:

On completion of this course, students will be able to :

- Design and apply iterative and recursive algorithms.
- Design and implement optimization algorithms in specific applications.
- Design appropriate shared objects and concurrent objects for applications.
- Develop and debug RPC based client-Server programs in UNIX.
- Realize the partial implementation of UDDI, SOAP, JMS in Web applications.

Text Book:

1. Sunitha Mahajan, Seema Shah: “Distributing Computing”, Oxford University press, 2010, ISBN-10: 0198061862, ISBN-13: 9780198061861, (Chapters 7-12,14).

Reference Book:

1. A.D. Kshemkalyani, M. Singhal: “Distributed Computing: Principles, Algorithms, and Systems”, Cambridge University Press, March 2011, ISBN: 9780521189842.

E-Resources:

1. https://global.oup.com/academic/product/distributed-computing-9780198093480
2. https://www.abebooks.com/book-search/author/sunita-mahajan-and-seema-shah/

Database Concepts (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI642	3-0-2-0	4	CIE:50 SEE:50	N/A	FE

Course Objectives:

This course will enable students to :

- Understand the basic concepts of database and Database Management System.
- Understand the difference between relational systems and non-relational systems.
- Get a clear understanding of how to maintain data (CRUD operations) in a relational database.
- Understand the working of a non-relational database.
- Get a clear understanding of how to maintain data (CRUD operations) in a non-relational database.
- Understand how Java programs can access database management systems using JDBC.

Syllabus

Module – I

Introduction to Database: Relational Database, Schema less Database.

Introduction to SQL: DDL, DML, DQL, TCL, DCL.

Constraints: Not null, Unique, Primary key, foreign key. **08 Hours**

Module – II

Operators and Functions: Operators and Functions.

Joins, Inner queries, Co-related queries: Joins, Inner queries, Co-related queries. **07 Hours**

Module – III

JDBC: JDBC API, Statement / Prepared Statements / CallableStatments, ResultSet, CRUD operations. **05 Hours**

Module – IV

MongoDB: Introduction and Installation, CRUD operations, Projections **06 Hours**

Module – V

Project Work: To create a Banking application using the concepts of database management systems with special emphasis on Java, JDBC, and MySQL database.

24 Hours

Laboratory

1. Programs covering SQL, Constraints, Operators and Functions, Joins, Inner queries, Co-related queries.
2. Programs covering JDBC concepts.
3. Programs covering MongoDB concepts.

Course Outcomes:

On completion of this course, students will be able to :

- Explain the basic concepts of database and Database Management System.
- Differentiate between relational systems and non-relational systems.
- Describe how to maintain data (CRUD operations) in relational and non-relational database.
- Manage Java programs to access database management systems using JDBC.
- Save and retrieve data in a safe and consistent manner.

Text Books:

1. Rajiv Chopra: "Database Management Systems (DBMS)", S Chand Publishing, 5th Edition, ISBN-10: 9385676342.
2. Kristina Chodorow: "MongoDB: The Definitive Guide", Shroff, 2nd Edition, ISBN-10: 9351102696.

Reference Books:

1. Raghu Ramakrishnan: "Database Management Systems" (Asia Higher Education Engineering/Computer Science), McGraw Hill Education, 3rd Edition; ISBN-10: 07123151X.
2. Kyle Banker, Peter Bakkum, Shaun Verch: "MongoDB in Action", Dreamtech Press, 2nd Edition, ISBN-10: 9351199355.

Computer Graphics and Multimedia (IC)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISI643	3:0:2:0	4	CIE:50 SEE:50	3 Hours	FE

Course Objectives:

This course will enable students to :

- Gain experience in interactive computer graphics using the OpenGL API.
- Perform 2D and 3D transformations and homogeneous co-ordinates.
- Enable students to acquire knowledge about Multimedia compression and animations.
- Learn Creation, Management and Transmission of Multimedia objects.

Syllabus

Module - I

Introduction: Applications of computer graphics; A graphics system:

Images: Physical and synthetic; Imaging Systems; The synthetic camera model; The programmer's Interface; Graphics architectures; Programmable Pipelines; Performance Characteristics. **08 Hours**

Module - II

Graphics Programming: The Sierpinski gasket, Programming two dimensional Applications, The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program; Polygons and recursion; **Input and Interaction:** Interaction; Input devices; Clients and Servers; Display Lists; Display Lists and Modeling; Programming Event Driven Input; Menus; Picking; **08 Hours**

Module - III

Geometric Objects and Transformations: Scalars, Points, and Vectors; Three-dimensional Primitives; Coordinate Systems and Frames; Modeling a Colored Cube; Affine Transformations; Rotation, Translation and Scaling; Geometric Objects and Transformations; Transformation in Homogeneous Coordinates; **08 Hours**

Module - IV

Introduction, Media and Data Streams, Audio Technology: Multimedia Elements; Multimedia Applications; Multimedia Systems Architecture; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Multimedia Data Interface Standards; The need for Data Compression; Multimedia Databases. Media:

Perception Media, Representation Media, Presentation Media, Storage Media, Transmission Media, Information Exchange Media, Presentation Spaces and Values, and Presentation Dimensions; Key Properties of a Multimedia System: Discrete and Continuous Media, Independence Media, Computer Controlled Systems, Integration; Characterizing Data Streams: Asynchronous Transmission Mode, Synchronous Transmission Mode, Isochronous Transmission Mode; Characterizing Continuous Media Data Streams. **08 Hours**

Module - V

Data Compression-1: Storage Space; Coding Requirements; Source, Entropy, and Hybrid Coding; Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT based Mode, Lossless Mode, Hierarchical Mode. **07 Hours**

List of Experiments

1. Write a program to recursively subdivide a tetrahedron to from 3D Sierpinski gasket. The number of recursive steps is to be specified by the user.
2. Write a program to implement Liang-Barsky line clipping algorithm.
3. Write a program to draw a color cube and spin it using OpenGL transformation matrices.
4. Write a program to create a house like figure and rotate it about a given fixed point using OpenGL functions.
5. Write a program to implement the Cohen-Sutherland line-clipping algorithm. Make provision to specify the input line, window for clipping and view port for displaying the clipped image.
6. Write a program using SCILAB that accepts an image (color or gray) and compresses it using the block truncation coding.

Course Outcomes:

On completion of this course, students will be able to :

- Get the concepts of Graphics display devices, different types of graphics drawing algorithms.
- Get the concepts of Viewing, Curves and surfaces.
- Understand concept of geometric, mathematical and algorithmic concepts necessary for programming computer graphics.

- Solve simple problems in the basic representation and handling of multimedia data (images, audio and animations).
- Design creative approach in application of multimedia devices, equipment and systems.

Text Books:

1. Edward Angel: "Interactive Computer Graphics A Top-Down Approach with OpenGL", 6th Edition, Pearson Education, 2008, ISBN-13: 978-0-13-254523-52, (Chapters 1, 2.1-2.8, 2.11,2.12,3.1-3.9).
2. Ralf Steinmetz, KlaraNarstedt: "Multimedia Fundamentals: Vol 1Media Coding and Content Processing", 2nd Edition, PHI, Indian Reprint 2008, ISBN: 9780132442435, (Chapters 1-3,7).

Reference Books:

1. Donald Hearn, Pauline Baker: "Computer Graphics- OpenGL Version", 4th Edition, Pearson Education, 2010.
2. F.S. Hill Jr.: "Computer Graphics Using OpenGL", 3rd Edition, PHI, 2009.
3. Prabhat K Andleigh, KiranThakrar: "Multimedia Systems Design", 1st Edition, PHI, 2007.

E- Resources:

1. http://www.tutorialspoint.com/computer_graphics.
2. https://www.tutorialspoint.com//multimedia
3. https://www.ddegjust.ac.in/studymaterial/mca-5.

Data Mining

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST651	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Understand the basic concepts, different types of data and data pre-processing methods.
- Understand the basic concepts and algorithms in association mining.
- Understand the different classification techniques.
- Analyze the accuracy of diverse classifiers and predictors.
- Identify the appropriate clustering techniques for the given data sets.
- Get familiarized with the various applications of Data Mining.

Syllabus

Module-I

Data Mining- Introduction, Challenges, Data Mining Tasks, Types of Data, Data Quality-Measurements and data collection errors, precision, bias, accuracy, missing value, inconsistent values. noise and artifacts, outliers, duplicate data, Data Preprocessing-aggregation, sampling, dimensionality reduction, discretization and binarization, variable transformation. **08 Hours**

Module-II

Association Analysis: Association Analysis- Basic Concepts and Algorithms: Frequent Item set Generation. Rule Generation, Compact Representation of Frequent Item sets, Alternative methods for generating Frequent Item sets, FP Growth Algorithm, Evaluation of Association Patterns–Objective Measures of Interestingness. **08 Hours**

Module-III

Classification: Basics, General approach to solve classification problem, Decision Trees Induction, Rule based classifiers, Nearest Neighbour Classifiers, Bayesian Classifiers, Artificial Neural Networks (ANN). **08 Hours**

Module-IV

Clustering Techniques: Overview, Features of cluster analysis, Types of Data and Computing Distance (Measures of Similarity andDissimilarity), Different types

of Clustering, Different types of clusters, K-means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation-Overview. **08 Hours**

Module-V

Mining different types of data: Text Mining-Text data analysis andinformational retrieval, text mining approaches, Mining the world wide web, Mining web page layout structure, Mining web's link structures, Web usage mining. Applications: Finance, Retail Industry, Intrusion detection. **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Identify the data and various data reprocessing techniques.
- Design association mining approaches and analyze them.
- Design data classification methods and measures for evaluation.
- Generate clusters from the data using similarity measures.
- Apply data mining approaches in text and web and identify the applications of data mining.

Text Book:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: "Introduction to Data Mining", (Chapters 1,2.1-2.3,4.1,5.1-5.4,6.1-6.7,8.1-8.5,10.3-10.5), Pearson Education, 2005, ISBN: 032-132-1367.

Reference Books:

1. Jiawei Han, Micheline-Kamber: "Data Mining - Concepts and Techniques", (Chapters 10.3-10.5), 2nd Edition, Morgan Kaufmann Publisher, 2006, ISBN: 978-81-312-0535-8.
2. G. K. Gupta: "Introduction to Data Mining with Case Studies", 3rd Edition, PHI, New Delhi, 2009.

E- Resources:

1. www-users.cs.umn.edu/~kumar/dmbook.
2. www.cs.ccsu.edu/~markov/ccsu_courses/datamining-1.html.
3. https://www.pearson.com/us/higher...Introduction-to-Data-Mining/PGM93748.html

Artificial Intelligence

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15IST652	3:0:0:0	3	CIE:50 SEE:50	3 Hours	EE

Course Objectives:

This course will enable students to :

- Get introduced to various search methods.
- Become familiar various knowledge representation methods.
- Provide the Symbolic Reasoning Under Uncertainty and Statistical Reasoning.
- Get familiarized with various Game Playing techniques.
- Get introduced to natural language processing using predicate logic.

Syllabus

Module – I

Introduction: What is AI?: The AI Problems, The Underlying Assumption, What Is An AI Techniques, The Level Of The Model, Criteria For Success, Some General References, One Final Word.

Problems, State Space Search and Heuristic Search Techniques: Defining The Problems As A State Space Search, Production Systems, Production Characteristics, Production System Characteristics, And Issues In The Design Of Search Programs, Additional Problems. Generate-And-Test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, Means-Ends Analysis. **08 Hours**

Module – II

Knowledge Representation and Logical Agents: Knowledge Representation Issues: Representations And Mappings, Approaches To Knowledge Representation.

Using Predicate Logic: Representation Simple Facts In Logic, Representing Instance And Isa Relationships, Computable Functions And Predicates, Resolution. Representing Knowledge Using Rules: Procedural Versus Declarative Knowledge, Logic Programming, Forward Versus Backward Reasoning. **08 Hours**

Module – III

Symbolic and Statictical Reasoning: Symbolic Reasoning Under Uncertainty: Introduction To Nonmonotonic Reasoning, Logics For Non-monotonic Reasoning. Statistical Reasoning: Probability And Bays' Theorem, Certainty Factors And Rule-Base Systems, Bayesian Networks, Dempster Shafer Theory, Fuzzy Logic. **08 Hours**

Module – IV

Game Playing: Overview, And Example Domain: Overview, MiniMax, Alpha-Beta Cut-off, Refinements, Iterative deepening, The Blocks World, Components Of A Planning System, Goal Stack Planning, Nonlinear Planning Using Constraint Posting, Hierarchical Planning, Reactive Systems, Other Planning Techniques. **08 Hours**

Module –V

Natural Language Processing: Introduction, Syntactic Processing, Semantic Analysis, Semantic Analysis, Discourse And Pragmatic Processing, Spell Checking.

Connectionist Models: Introduction: Hopfield Network, Learning In Neural Network, Application Of Neural Networks, Recurrent Networks, Distributed Representations, Connectionist AI And Symbolic AI. **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Describe and Understand various search methods.
- Differentiate various knowledge representation methods.
- Solve issues Symbolic Reasoning Under Uncertainty and Statistical Reasoning.
- Manage various Game Playing techniques.
- Design natural language processing using predicate logic.

Text Books:

1. Elaine Rich, Kevin Knight: "Artificial Intelligence", 2nd Edition, Tata Mcgraw-Hill, 2009, ISBN: 07087709, ID: 7749235760.
2. Stuart Russel, Peter Norvig: "Artificial Intelligence: A Modern Approach", PHI, ISBN-10: 0-13-604259-7, ISBN-13: 978-0-13-604259-4.

Reference Books:

1. Stuart Russel, Peter Novig: "AI – A Modern Approach", 2nd Edition, Pearson Education, 2007.
2. Deepak Khemani: "Artificial Intelligence", Tata Mc-Graw Hill Education, 2013.

List of Open Source Software/learning website:

1. http://www.journals.elsevier.com/artificial-intelligence/
2. https://www.technologyreview.com/s/534871/our-fear-of-artificial-intelligence/
3. http://www.sanfoundry.com/artificial-intelligence-mcqs-inductive-logic-unification-lifting-1/.

LabVIEW - Level I

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE661	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

Course Objectives:

This course will enable students to :

- Understand the fundamental of graphical coding system.
- Learn to develop basic level of LabVIEW coding.
- Study the different component of LabVIEW operating tools.
- Study and develop state machine for a specific problem.
- Develop integrated coding solution for analysis and presentation with MyRio hardware using accelerometer.

Syllabus

Module - I

LabVIEW programming concepts, environment and Software constructs: Data flow, Polymorphism, Front panel window, block diagram, and connector pane, Menus and palettes, Configuration options. Controls, indicators, IO controls, and refnums Terminals, constants, nodes, update modes, and legends of charts and graphs. Mechanical action of Boolean objects Property Nodes. Numeric, string, Boolean, and path data types. Array and cluster data types. Shift registers, Case, Sequence and Event structures. **10 Hours**

Module - II

Programming, Data communication and synchronization VIs and functions: Conversion, comparison, and manipulation, Timing and Timing functions related to Timed structures. Data storage and file I/O formats, Waveform and waveform file I/O, Dynamic and User events Local, global, and shared variables Data Socket TCP and UDP Notifiers Queues Semaphores Property Nodes, and Invoke Nodes. **08 Hours**

Module - III

Error handling VIs and functions: Error clusters Dialog and User Interface VIs Custom error codes.

Design patterns: Simple state machine, User interface event handler, Queued message handler, producer/consumer (data) and producer/consumer (events), Functional global variables. **06 Hours**

Module - IV

Sub VI design: SubVI creation methods, Connector panes and connection types, Polymorphic subVIs, Options related

Debugging tools and techniques: Debugging tools, Error list window, Execution highlighting, Breakpoints and single stepping, Generic and custom probes, Debugging practices and techniques for different situations. **08 Hours**

Module - V

VI design and documentation (style) practices: Refer to the LabVIEW Style Checklist topic of the LabVIEW Help for information on the following items

i. User interface design and block diagram layout

ii. Modular and hierarchical design

iii. SubVI icons and connector pane layout (standard)

iv. Properties

v. Documenting Vis

Memory, performance, and determination

a. Tools for identifying memory and performance issues

Profile memory and performance, Show buffer allocations and VI metrics

b. Programming practices

Enforcing dataflow, User interface updates and response to user interface controls, Data type selection, coercion, and buffer allocation, Array, string, and loop operations **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Formulate basic aspects of the graphical programming using LabVIEW 2016.
- Develop LabVIEW coding for a specific problem of datalogging, measurement and presentation.
- Handle the error function and errors in the LabVIEW coding.
- Develop coding for data handling and Analysis on the acquired data.
- Design a state machine LabVIEW coding for an applied problem.

Text Books:

1. "LabVIEW - Getting Started with LabVIEW", M/s National Instruments, 2013 373427J-01.
2. Jovitha Jerome: "Virtual instrumentation using labview", PHI Learning Pvt. Ltd., 2010.
3. Hans-Petter Halvorsen: "Introduction to LabVIEW," University College of South-east, Norway.
4. S. Sumathi, P. Surekha: "LabVIEW based Advanced Instrumentation Systems", Springer.
5. Lab manual provided by Dept. of Civil Engg., NCET.

Reference Books:

1. Jeffrey Travis, Jim Kring: "Introduction to Graphical Programming with LabVIEW", Pearson, 2006.
2. Malan Shiralkar: "LabVIEW Graphical Programming Course Collection", National Instruments.

E-Resources:

1. http://cnx.org/content/col10241/1.4.

Yoga and Meditation

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE662	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

Course Objectives:

This course will enable students to :

- Introduce the main principles of Yoga.
- Generate knowledge and skills of students to use the tools and techniques for using Yoga in day to day life for better health and well being.
- Improve communication and increase concentration through Yoga and Meditation.
- Equip the individual to handle stressful situations and manage day to day activities.

Syllabus

Module – I

Definition and meaning of yoga: Meaning of Asanas, Types of Asanas: standing, sitting and supine asanas. Standing Asanas (Trikon asan, padhastasan, ardhachakrasan, veerbhadrasan), Sitting Asanas (Vajrasan, padmasan, suptavajrasan, Ardhamaschendrasan, vakrasan), Supine Asanas (Sarvangasan, Matsyasan, Natarajasan, Shavasan) **08 Hours**

Module – II

Patanjali's Yoga Sutra: Eight limbs of yoga, Importance of discipline in Yoga, Stillness of mind, Five Modulations (vritti) of the mind, Practice and Dispassion, Obstacles in the path of Yoga, Overcoming distractions of the mind through Yoga. **08 Hours**

Module – III

Understanding physiological implications of Yoga, Three types of Gunas (Satva, Rajas and Tamas) and their effects on body and mind, Food Habits, Meaning of Prana, Pranayama and its advantages, Different types of Pranayama. **08 Hours**

Module – IV

Ayurveda: The science of life, Three types of doshas (Vata, Pitta and Kapha), Balancing the different doshas for a healthy life, Ayurvedic principles of food and activity, Advanced Asanas: Mayurasana, Sirsasana, Gomukh Asana, Vrkshasana, Baddha Konasana. **08 Hours**

Module – V

Meditation: Meaning of meditation, Meditation vs Concentration, Advantages of Meditation, Effects of Meditation on body and mind, Effect on health and general well being, Reducing stress through meditation, Increasing concentration, Improving communication, Effect on Environment **08 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Know the basic principles of Yoga.
- Know and practice the basic asanas and their benefits.
- Use Pranayama and Meditation for improving health and mental peace.
- Know the difference between meditation and concentration.
- Apply the principles of Ayurveda and implement them for one's benefit.

Text Books:

1. Yoga Sutras of Patanjali (ancient text).
2. B K S Iyengar: "Light on Yoga".

Reference Books:

1. A traditional touch to Yogasanas for beginners and Sadhakas, Swami Vivekananda Yoga Prakashana (SVYP).
2. Dr. Vasant Lad: "Ayurveda: The Science of Self-Healing: A Practical Guide".

ॐ ॐ ॐ ॐ ॐ

Martial Arts

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE663	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

The following types of Martial arts are offered,

1. Karate
2. Taekwondo
3. Judo
4. Kung-fu

Expert Trainers will be provided during the academic year through experts in Martial Arts. Students who enroll for this elective should attend the regular Training classes and maintain a minimum of 85% attendance.

At the end of the training programme the performance Evaluation will be made by team of experts. Students who secure at least a satisfactory grade will be issued a certificate and deemed to have been completed the above said 3 Credit course. However, the students who have shortage of attendance will be consider for the award of 3 credits provided they undergo training at any of the training centers in the above said Martial Arts, complete the certification programme and give a demo along with viva in the presence of experts in the campus.

ଓ ଓ ଓ ଓ ଓ

Music (Carnatic Vocal/Instrumental)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE664	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

Course Objectives:

This course will enable students to :

- Get familiarized with the conceptual understanding of Carnatic music.
- Gain knowledge about the basics of Swaravalis.
- Understand the use of different Talas.
- Gain understanding about various Raagas.
- Gain understanding about intricacies of Swaras.

Syllabus

Module – I

Theoretical Aspects: Father of Carnatic music, Famous personalities in Carnatic music, Concept of Sapta Swara, Taala, Melody, Pitch, Rhythm, Janaka Raaga, Janya Raaga. **03 Hours**

Module – II

Sarale Varase (Any 5), Janti Varase (Any 5), Daatu Varase, Tara Stayi, Mandra Stayi. **08 Hours**

Module – III

Alankaras: Druva Taala, Matya Taala, Triputa Taala, Rupaka Taala, Jampe Taala, Atta Taala, Eka Taala. **08 Hours**

Module – IV

Geethagalu, Pillari Geethe (4), Sanchari Geethe (5), Lakshana Geethe (1). **10 Hours**

Module – V

Swarajatis (Any 2), Kalyani, Bilahari, Neelambari, Kamach.

Varna (Any 2), Shankarabarana, Kalyani, Hamsadwani , Mohana. **10 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Gain knowledge about the theoretical background of carnatic music
- Acquire practical knowledge on basics of Carnatic music.
- Practical demonstration of different Talas.
- Distinguish among various Raagas based on swara sthanas.
- To synchronize the Raaga and Taala.

Text books:

1. Dr. Sachidevi: "Karnataka Sangeetha Darpana", Sreenivasa Prakashana, Bengaluru, 2014.
2. Junior Carnatic Music – C Shiva Musicals, Malleshwaram, Bengaluru, 2013.

ଓ ଓ ଓ ଓ ଓ

Dance (Bharatanatya)

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE665	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

Course Objectives:

This course will enable students to :

- Gain theoretical knowledge about various types of Indian dances.
- Understand about various musical instruments used in Bharatanatya.
- Learn Practical demonstrations of Bharatanatya steps on Prarthane Namaskara and Shlokas.
- Learn the movements of head, neck, eyes, hands according to Bharatanatya steps.
- Learn the brisk movements in Bharatanatya with the help of ADAVUS.

Syllabus

Module – I

Indian Classical dance, It's history and Significance, Types of Classical Dance, Bharatanatya, Kathakali, Mohini Attam, Koochipudi, Katahak, Odissi, Manipuri.

04 Hours

Module – II

Musical Instuments used in Bharatanatya: Tabala, Mrudanga, Kamsale, Kolata, Taala vadya. Famous personalities in Bharatanatya, Composers of Natya Grantas.

03 Hours

Module – III

Practical exercises on Prarthane, Namaskara and Shloka, Vyayama Kriye for Bharatanatya (Two Shlokas and Two Prarthanes). **10 Hours**

Module – IV

Abhinaya Steps (Chaturvidha) ShiroBedha, Drushti Bedha, Greeva Bedha, Brubedha, Hasta Bedha (Samyuta and Asamyuta). **10 Hours**

Module – V

Adavugalu (DashaVidha) Tattu adavu, Mettu Adavu, Nat Adavu, Egaru Tattu Adavu, Egaru Mettu adavu, Jaaru Adavu, Mandi adavu, TattuMettu Adavu, Rangakarma Adavu, Teermana Adavu. **12 Hours**

Course Outcomes:

On completion of this course, students will be able to :

- Get an insight into various types of Indian dances.
- Gain knowledge of different instruments used to perform dance.
- Perform exercises on prarthane, Namaskara according to Bharatanatya style.
- Perform basic steps in Abhinaya.
- Recognise and perform different Adavus.

Reference Book:

1. "Bharatanatya shastra", Department of Public Instruction, Karnataka State Government**.**

Sports

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15HOE666	2:0:0:4	3	CIE:50 SEE:50	3 Hours	OE

Students who are selected by the University to represent the VTU teams and for participating at State level / National level Sports in the following sports are exempted from taking open elective (Code:15HOE666) and will be awarded 3 credits.

Outdoor games	Indoor games
Cricket	Carrom
Foot ball	Chess
Hockey	Shuttle Badminton
Basket Ball	Squash
Kabbadi	Table – Tennis
Kho – Kho	Gymnastics
Hand – Ball	
Athletics	
Swimming	
Lawn Tennis	

The achievement in Sports as said above should have been made during the academic year during which the said open elective is offered.

After representing at VTU / State / National level in any of the above said sports, the students should produce the certificates from the competent authorities. Based on the certificates the institution will issue another certificate related to the achievement and awarding of three credits.

ଓ ଓ ଓ ଓ ଓ

Unix System Programming Laboratory

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15ISL67	1:0:2:0	2	CIE:50 SEE:50	3 Hours	FC

Course Objectives:

This course will enable students to :

- Have hands-on experience on the theoretical concepts through simple experiments.
- Use the different manifested constants to determine the system is POSIX conforming.
- Use API's to implement interposes communication and other basic services of UNIX kernel.
- Use system calls to create processes that manipulate system resources and get the system configuration limits.

Syllabus

List of Experiments for USP:

Design, develop, and execute the following programs,

1. Write a C/C++ POSIX compliant program to check the following limits:
   i. No. of clock ticks.
   ii. Max. No. of child processes.
   iii. Max. Path length.
   iv. Max. No. of characters in a file name.
   v. Max. No. of open files/ process.
2. Write a C/C++ POSIX compliant program that prints the POSIX defined configuration options supported on any given system using feature test macros.
3. Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.
4. a) Write a C/C++ program that outputs the contents of its Environment list.
   b) Write a C / C++ program to emulate the UNIX **ln** command.
5. Write a C/C++ program that creates a zombie and then calls system to execute the **ps**command to verify that the process is zombie.
6. Write a C/C++ program to implement the **system** function.

7. Write a C/C++ program to set up a real-time clock interval timer using the **alarm** API.
8. C/C++ program to do the following using fork, create a child process, the child process prints its own process id and the id of its parents and then exit. Parent process waits for its child to finish and it prints its own id and its child process id and then exits.
9. C/C++ program that accepts valid filename as a command line argument and for each of the argument prints the type of the file.
10. C/C++ program uses access API to determine command line argument, whether a named file exists. If named file exist program will read data from the file. If named file does not exist, it will create and string "hello world" will be written to the named file.
11. C/C++ program that takes the file name as command line argument and modify the access, modification time stamp of the file.

Course Outcomes:

On completion of this course, students will be able to :

- Analyze the different manifested constants used to determine the system is configurations.
- Use API's to implement interposes communication and other basic services of UNIX kernel.
- Design programs to create processes that manipulate system resources and get the system configuration limits.
- Develop programs to lock the file and record.
- Demonstrate how to set up a real-time clock interval timer using the **alarm** API.

Text Books:

1. Terrence Chan: "UNIX System Programming Using C++", Prentice Hall India, 1999, ISBN-10: 0133315622, (Listed topics only from Chapters 1,5-10).
2. W. Richard Stevens: "Advanced Programming in the UNIX Environment", 2nd Edition, Pearson Education, 2005, ISBN: 0201433079, (Listed topics only from Chapters 7-9,13-15).

E-Resources:

1. www.compsci.hunter.cuny.edu/~sweiss/course_materials/unix_lecture.../chapter_01.pdf
2. www.cs.northwestern.edu/~pdinda/netclass-f00/unix_nutshell.pdf

Technical Aptitude and GD

Course Code	L:T:P:S	Credits	Exam marks	Exam Duration	Course Type
15CSH68	2:0:0:0	2	CIE:50 SEE:50	3 Hours	HSS

The respective branches shall conduct training programmes related to important and latest programming languages and other emerging technologies, such as Solar and Electric power based gadgets, IoT, ROBOT's, Environmental friendly and cost effective construction techniques, UAV's and technologies pertaining to the respective department.