# B-Tech Computer Science & Engineering

## Fifth Semester

## Course Syllabus & Course Plan

# S5 - CORE COURSES

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS301 | THEORY OF COMPUTATION | 3-1-0-4 | 2015 |

**Course Objectives**

1. *Introduce the concept of formal languages.*
2. *Discuss the Chomsky classification of formal languages with discussion on grammar and automata for regular, context-free, context sensitive and unrestricted languages.*
3. *Discuss the notions of decidability and halting problem.*

**Syllabus**

Introduction to Automata Theory, Structure of an automaton, classification of automata, grammar and automata for generating each class of formal languages in the Chomsky Hierarchy, decidability and Halting problem.

**Expected Outcome**

Student is able to

1. *Classify formal languages into regular, context-free, context sensitive and unrestricted languages.*
2. *Design finite state automata, regular grammar, regular expression and Myhill- Nerode relation representations for regular languages.*
3. *Design push-down automata and context-free grammar representations for context-free languages.*
4. *Design Turing Machines for accepting recursively enumerable languages.*
5. *Understand the notions of decidability and undecidability of problems, Halting problem.*

**Text Books**

1. John E Hopcroft, Rajeev Motwani and Jeffrey D Ullman, Introduction to Automata Theory, Languages, and Computation, 3/e, Pearson Education, 2007
2. John C Martin, Introduction to Languages and the Theory of Computation, TMH, 2007
3. Michael Sipser, Introduction To Theory of Computation, Cengage Publishers, 2013

**References**

1. Dexter C. Kozen, Automata and Computability, Springer1999.

| Course Plan | | | |
|---|---|---|---|
| **Module** | **Contents** | **Hours** | **Sem. Exam Marks %** |
| **I** | Introduction to Automata Theory and its significance. **Type 3 Formalism:** Finite state automata – Properties of transition functions, Designing finite automata, NFA, Finite Automata with Epsilon Transitions, Equivalence of NFA and DFA, Conversion of NFA to DFA, Equivalence and Conversion of NFA with and without Epsilon Transitions. | **09** | **15 %** |
| **II** | Myhill-Nerode Theorem, Minimal State FA Computation. Finite State Machines with Output- Mealy and Moore machine (Design Only), Two- Way Finite Automata. Regular Grammar, Regular Expressions, Equivalence of regular expressions and NFA with epsilon transitions. Converting Regular Expressions to NFA with epsilon transitions Equivalence of DFA and regular expressions, converting DFA to Regular Expressions. | **09** | **15 %** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Pumping Lemma for Regular Languages, Applications of Pumping Lemma. Closure Properties of Regular sets (Proofs not required), Decision Problems related with Type 3 Formalism **Type 2 Formalism:-** Context-Free Languages (CFL), Context-Free Grammar (CFG), Derivation trees, Ambiguity, Simplification of CFG, Chomsky Normal Form, Greibach normal forms | **08** | **15 %** |

| | | | |
|---|---|---|---|
| **IV** | Non-Deterministic Pushdown Automata (NPDA), design. Equivalence of acceptance by final state and empty stack in PDA. Equivalence between NPDA and CFG, Deterministic Push Down Automata, Closure properties of CFLs (Proof not required), Decision Problems related with Type 3 Formalism. | **07** | **15 %** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Pumping Lemma for CFLs, Applications of Pumping Lemma.<br>**Type 1 Formalism:** Context-sensitive Grammar. Linear Bounded Automata (Design not required)<br>**Type 0 Formalism:** Turing Machine (TM) – Basics and formal definition, TMs as language acceptors, TMs as Transducers, Designing Turing Machines. | **08** | **20 %** |
| **VI** | Variants of TMs -Universal Turing Machine, Multi-tape TMs, Non Deterministic TMs, Enumeration Machine (Equivalence not required), Recursively Enumerable Languages, Recursive languages, Properties of Recursively Enumerable Languages and Recursive Languages, Decidability and Halting Problem. Chomsky Hierarchy | **07** | **20 %** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18

b. *Three*questions each having 9 marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts.

4. Part C
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS303 | SYSTEM SOFTWARE | 2-1-0-3 | 2015 |

**Course Objectives**

*To make students understand the design concepts of various system software like Assembler, Linker, Loader and Macro pre-processor, Utility Programs such as Text Editor and Debugger.*

**Syllabus**

Different types of System Software, SIC & SIC/XE Architecture and Programming, Basic Functions of Assembler, Assembler Design, Single pass and 2 Pass Assemblers and their Design, Linkers and Loaders, Absolute Loader and Relocating loader, Design of Linking Loader, Macro Processor and its design, Fundamentals of Text Editor Design, Operational Features of Debuggers.

**Expected Outcome**

Student is able to

1. *Distinguish different software into different categories.*
2. *Analyze, design and implement one pass, two pass or multi pass assembler.*
3. *Analyze, designand implement loader and linker.*
4. *Analyze, designand implement macro processors.*
5. *Critique the features of modern editing /debugging tools.*

**Text book**

1. Leland L. Beck, System Software: An Introduction to Systems Programming, 3/E, Pearson Education Asia.

**References**

1. D.M. Dhamdhere, Systems Programming and Operating Systems, Second Revised Edition, Tata McGraw Hill.
2. John J. Donovan, Systems Programming, Tata McGraw Hill Edition 1991.
3. Peter Abel, IBM PC Assembly Language and Programming, Third Edition, Prentice Hall of India.
4. J. Nithyashri, System Software, Second Edition, Tata McGraw Hill.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | ***Introduction***<br>System Software Vs. Application Software, Different System Software– Assembler, Linker, Loader, Macro Processor, Text Editor, Debugger, Device Driver, Compiler, Interpreter, Operating System(Basic Concepts only)<br><br>SIC & SIC/XE Architecture, Addressing modes, SIC & SIC/XE Instruction set, Assembler Directives and Programming. | **02**<br><br>**06** | **15%** |
| II | ***Assemblers***<br>Basic Functions of Assembler. Assembler output format – Header, Text and End Records- Assembler data structures, Two pass assembler algorithm, Hand assembly of SIC/XE program, Machine dependent assembler features. | **06** | **15 %** |
| **FIRST INTERNAL EXAM** | | | |
| III | ***Assembler design options***<br>Machine Independent assembler features – program blocks, Control sections, Assembler design options- Algorithm for Single Pass assembler, Multi pass assembler, Implementation example of MASM Assembler | **07** | **15%** |
| IV | ***Linker and Loader***<br>Basic Loader functions - Design of absolute loader, Simple bootstrap Loader, Machine dependent loader features- Relocation, Program Linking, Algorithm and data structures of two pass Linking Loader, Machine dependent loader features, Loader Design Options. | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |

| | | | |
|---|---|---|---|
| **V** | ***Macro Preprocessor***<br>Macro Instruction Definition and Expansion. One pass Macro processor Algorithm and data structures, Machine Independent Macro Processor Features, Macro processor design options | **07** | **20%** |
| **VI** | ***Text Editors***<br>Overview of Editing, User Interface, Editor Structure.<br>***Debuggers***<br>Debugging Functions and Capabilities, Relationship with other parts of the system, Debugging Methods- By Induction, Deduction and Backtracking. | **02**<br><br>**04** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having *3* marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three* questionseach having *9* marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
    a. Total marks : 12
    b. *Four* questions each having *3* marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18

b. *Three* questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts

6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| CS305 | Microprocessors and Microcontrollers | 2-1-0-3 | 2015 |

**Pre-requisites**

1. **BE 101-05** Introduction to Computing and Problem Solving
2. **CS203** Switching Theory and Logic Design
3. **CS202** Computer Organization and Architecture

**Course Objectives**

1. *To impart basic understanding of the internal organisation of 8086 Microprocessor and 8051 microcontroller.*
2. *To introduce the concepts of interfacing microprocessors with external devices.*
3. *To develop Assembly language programming skills.*

**Syllabus**

Introduction to 8086 Microprocessor; Architecture and signals, Instruction set of 8086, Timing Diagram, Assembly Language Programming, Memory and I/O interfacing, Interfacing with 8255, 8279, 8257, Interrupts and Interrupt handling, Microcontrollers - 8051 Architecture and its salient features, Instruction Set and Simple Programming Concepts.

**Expected Outcome**

Student is able to

1. *Describe different modes of operations of a typical microprocessor and microcontroller.*
2. *Design and develop 8086 assembly language programs using software interrupts and various assembler directives.*
3. *Interface microprocessors with various external devices.*
4. *Analyze and compare the features of microprocessors and microcontrollers.*
5. *Design and develop assembly language programs using 8051 microcontroller.*

**Text Books**

1. Bhurchandi and Ray, *Advanced Microprocessors and Peripherals*, Third Edition McGraw Hill.
2. Raj Kamal, *Microcontrollers: Architecture, Programming, Interfacing and System Design*, Pearson Education.
3. Douglas V. Hall, SSSP Rao, *Microprocessors and Interfacing*, Third Edition, McGrawHill Education.

**References**

1. Barry B. Brey, *The Intel Microprocessors – Architecture, Programming and Interfacing*, Eigth Edition, Pearson Education.
2. A. NagoorKani, *Microprocessors and Microcontrollers*, Second Edition, Tata McGraw Hill.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| **I** | Evolution of microprocessors, 8086 Microprocessor - Architecture and signals, Memory organisation, Minimum and maximum mode of operation, Minimum mode Timing Diagram. Comparison of 8086 and 8088. | **07** | **15%** |
| **II** | 8086 Addressing Modes, 8086 Instruction set and Assembler Directives - Assembly Language Programming with Subroutines, Macros, Passing Parameters, Use of stack. | **08** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Interrupts - Types of Interrupts and Interrupt Service Routine. Handling Interrupts in 8086, Interrupt programming. Basic Peripherals and their Interfacing with 8086 - Programmable Interrupt Controller - 8259 - Architecture. | **07** | **15%** |

| | | | |
|---|---|---|---|
| **IV** | Interfacing Memory, I/O, 8255 - Detailed study - Architecture, Control word format and modes of operation, Architecture and modes of operation of 8279 and 8257 (Just mention the control word, no need to memorize the control word format) | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Microcontrollers - Types of Microcontrollers - Criteria for selecting a microcontroller - Example Applications. Characteristics and Resources of a microcontroller. Organization and design of these resources in a typical microcontroller - 8051.<br>8051 Architecture, Register Organization, Memory and I/O addressing, Interrupts and Stack. | **08** | **20%** |
| **VI** | 8051 Addressing Modes, Different types of instructions and Instruction Set, Simple programs.<br>Peripheral Chips for timing control - 8254/8253. | **08** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three*questions each having 9 marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12

b. *Four* questions each having 3 marks, uniformly covering modules III and IV;All*four* questions have to be answered.

5. Part D
   a. Total marks : 18
   b. *Three*questionseach having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS307 | DATA COMMUNICATION | 3-0-0-3 | 2015 |

**Course Objectives**

1. *To introduce fundamental communication models.*
2. *To discuss various time domain and frequency domain concepts of data communication.*
3. *To introduce the concepts of encoding, multiplexing and spread spectrum.*

**Syllabus**

Data Transmission, Transmission Impairments, Channel Capacity, Transmission media, Wireless propagation, Signal encoding Techniques, Multiplexing, Digital data transmission techniques, Sampling theorem, Error detection and correction, Spread spectrum, Basic principles of switching.

**Expected Outcome**

Student is able to

1. *Identify and list the various issues present in the design of a data communication system.*
2. *Apply the time domain and frequency domain concepts of signals in data communication.*
3. *Compare and select transmission media based on transmission impairments and channel capacity.*
4. *Select and use appropriate signal encoding techniques and multiplexing techniques for a given scenario.*
5. *Design suitable error detection and error correction algorithms to achieve error free data communication and explain different switching techniques.*

**Text Books**

1. William Stallings, Data and Computer Communication 9/e, Pearson Education, Inc. [Chapters: 4, 5, 6, 7, 8, 9].
2. Forouzan B. A., Data Communications and Networking, 5/e, Tata McGraw Hill, 2013. [Chapters:3,4, 5, 6,7,8]
3. Schiller J., Mobile Communications, 2/e, Pearson Education, 2009. [Chapters:2,3]
4. Curt M. White, Fundamentals of Networking and Communication 7/e, Cengage learning. [Chapter 3,4,9,10]

**References**

1. Forouzan B. A., Data Communications and Networking, 4/e, Tata McGraw Hill, 2007.
2. Tanenbaum A. S. and D. Wetherall, Computer Networks, Pearson Education, 2013.

**COURSE PLAN**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| **I** | Data Transmission: Communication model Simplex, half duplex and full duplex transmission - Periodic Analog signals: Sine wave, phase, wavelength, time and frequency domain, bandwidth - Digital Signals; Digital data Transmission:- Analog & Digital data, Analog & Digital signals, Analog &Digital transmission – Transmission Impairments: Attenuation, Delay distortion, Noise - Channel capacity: Nyquist Bandwidth, Shannon's Capacity formula. | **08** | **15%** |
| **II** | Transmission media - Guided Transmission Media: Twisted pair, Coaxial cable, optical fiber, Wireless Transmission, Terrestrial microwave, Satellite microwave. Wireless Propagation: Ground wave propagation, Sky Wave propagation, LoS Propagation. | **07** | **15%** |
| **FIRST INTERNAL EXAM** | | | |

| | | | |
|---|---|---|---|
| **III** | Signal Encoding techniques - Digital Data Digital Signals: NRZ, Multilevel binary, Biphase - Digital Data Analog Signals : ASK, FSK, PSK - Analog Data Digital Signals: Sampling theorem, PCM, Delta Modulation - Analog Data Analog Signals: AM, FM, PM. | **07** | **15%** |
| **IV** | Multiplexing- Space Division Multiplexing- Frequency Division Multiplexing: Wave length Division Multiplexing - Time Division multiplexing: Characteristics, Digital Carrier system, SONET/SDH-Statistical time division multiplexing: Cable Modem - Code Division Multiplexing. Multiple Access– CDMA. | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Digital Data Communication Techniques - Asynchronous transmission, Synchronous transmission-Detecting and Correcting Errors-Types of Errors-Error Detection: Parity check, Cyclic Redundancy Check (CRC) - Error Control Error Correction: Forward Error Correction and Hamming Distance. | **06** | **20%** |
| **VI** | Spread Spectrum Techniques-Direct Sequence Spread Spectrum (DSSS), Frequency Hopping Spread Spectrum (FHSS).<br>Basic principles of switching - Circuit Switched Networks, Structure of Circuit Switch - Packet Switching: Datagram Networks, Virtual Circuit Networks, Structure of packet switches. | **07** | **20%** |

| | | | |
|---|---|---|---|
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II;All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules III and IV;All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three*questionseach having 9 marks, uniformly covering modules III and IV;*Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS309 | GRAPH THEORY AND COMBINATORICS | 2-0-2-3 | 2015 |

**Course Objectives**

*To introduce the fundamental concepts in graph theory, including properties and characterization of graphs/ trees and Graphs theoretic algorithms*

**Syllabus**

Introductory concepts of graphs, Euler and Hamiltonian graphs, Planar Graphs, Trees, Vertex connectivity and edge connectivity, Cut set and Cut vertices, Matrix representation of graphs, Graphs theoretic algorithms.

**Expected Outcome**

Student is able to

1. *Demonstrate the knowledge of fundamental concepts in graph theory, including properties and characterization of graphs and trees.*
2. *Use graphs for solving real life problems.*
3. *Distinguish between planar and non-planar graphs and solve problems.*
4. *Develop efficient algorithms for graph related problems in different domains of engineering and science.*

**Text Books**

1. NarasinghDeo, Graph theory, PHI.
2. Douglas B. West, Introduction to Graph Theory, Prentice Hall India Ltd.
3. Robin J. Wilson, Introduction to Graph Theory, Longman Group Ltd.

**References**

1. R. Diestel,Graph Theory, free online edition: diestel-graph-theory.com/basic.html.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | **Introductory concepts** - What is graph – Application of graphs – finite and infinite graphs – Incidence and Degree – Isolated vertex, pendent vertex and Null graph. Paths and circuits – Isomorphism, sub graphs, | 09 | 15 % |

| | | | |
|---|---|---|---|
| | walks, paths and circuits, Connected graphs, disconnect graphs. | | |
| **II** | Euler graphs, Hamiltonian paths and circuits, Dirac's theorem for Hamiltonicity, Travelling salesman problem. Directed graphs – types of digraphs, Digraphs and binary relation | **10** | **15 %** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Trees – properties, pendent vertex, Distance and centres - Rooted and binary tree, counting trees, spanning trees. | **07** | **15 %** |
| **IV** | Vertex Connectivity, Edge Connectivity, Cut set and Cut Vertices, Fundamental circuits, Planar graphs, Different representation of planar graphs, Euler's theorem, Geometric dual, Combinatorial dual. | **09** | **15 %** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Matrix representation of graphs- Adjacency matrix, Incidence Matrix, Circuit matrix, Fundamental Circuit matrix and Rank, Cut set matrix, Path matrix | **08** | **20 %** |
| **VI** | Graphs theoretic algorithms - Algorithm for computer representation of a graph, algorithm for connectedness and components, spanning tree, shortest path. | **07** | **20 %** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B

a. Total marks : 18
b. *Three*questions each having *9* marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts.

4. Part C
   a. Total marks : 12
   b. *Four* questions each having *3* marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three*questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts.
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical questions.

# S5 - ELECTIVES

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS361 | SOFT COMPUTING | 3-0-0-3 | 2015 |

**Course Objectives**

*To introduce the concepts in Soft Computing such as Artificial Neural Networks, Fuzzy logic-based systems, genetic algorithm-based systems and their hybrids.*

**Syllabus**

Introduction to Soft Computing, Artificial Neural Networks, Fuzzy Logic and Fuzzy systems, Genetic Algorithms, hybrid systems.

**Expected Outcome**

Student is able to

1. *Learn about soft computing techniques and their applications.*
2. *Analyze various neural network architectures.*
3. *Define the fuzzy systems.*
4. *Understand the genetic algorithm concepts and their applications.*
5. *Identify and select a suitable Soft Computing technology to solve the problem; construct a solution and implement a Soft Computing solution.*

**Text Books**

1. S. N. Sivanandam and S. N.Deepa, Principles of soft computing - Wiley India.
2. Timothy J. Ross, Fuzzy Logic with engineering applications – Wiley India.

**References**

1. N. K. Sinha and M. M. Gupta, Soft Computing & Intelligent Systems: Theory & Applications-Academic Press /Elsevier. 2009.
2. Simon Haykin, Neural Network- A Comprehensive Foundation- Prentice Hall International, Inc.
3. R. Eberhart and Y. Shi, Computational Intelligence: Concepts to Implementation, Morgan Kaufman/Elsevier, 2007.
4. Ross T.J. , Fuzzy Logic with Engineering Applications- McGraw Hill.
5. Driankov D., Hellendoorn H. and Reinfrank M., An Introduction to Fuzzy Control-Narosa Pub.
6. Bart Kosko, Neural Network and Fuzzy Systems- Prentice Hall, Inc., Englewood Cliffs
7. Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning-

| | | | |
|---|---|---|---|
| Addison Wesley. | | | |
| Course Plan | | | |
| Module | Contents | Hours | Sem. Exam Marks % |
| **I** | Introduction to Soft Computing<br>Artificial neural networks - biological neurons, Basic models of artificial neural networks – Connections, Learning, Activation Functions, McCulloch and Pitts Neuron, Hebb network. | **08** | **15%** |
| **II** | Perceptron networks – Learning rule – Training and testing algorithm, Adaptive Linear Neuron, Back propagation Network – Architecture, Training algorithm | **08** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Fuzzy logic - fuzzy sets - properties - operations on fuzzy sets, fuzzy relations - operations on fuzzy relations | **07** | **15%** |
| **IV** | Fuzzy membership functions, fuzzification, Methods of membership value assignments – intuition – inference – rank ordering, Lambda –cuts for fuzzy sets, Defuzzification methods | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Truth values and Tables in Fuzzy Logic, Fuzzy propositions, Formation of fuzzy rules - Decomposition of rules – Aggregation of rules, Fuzzy Inference Systems - Mamdani and Sugeno types, Neuro-fuzzy hybrid systems – characteristics - classification | **08** | **20%** |
| **VI** | Introduction to genetic algorithm, operators in genetic algorithm - coding - selection - cross over – mutation, Stopping condition for genetic algorithm flow, Genetic-neuro hybrid systems, Genetic-Fuzzy rule based system | **08** | **20%** |
| **END SEMESTER EXAMINATION** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four*questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three sub-parts
4. Part C
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules III and IV;All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical/design questions.

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS363 | Signals and Systems | 3-0-0-3 | 2015 |

**Pre-requisites**

**NIL**

**Course Objectives**

1. *To introduce fundamental concepts of continuous time and discrete time signals.*
2. *To introduce fundamental concepts of continuous time and discrete time systems.*
3. *To introduce frequency domain representation and analysis of signals.*

**Syllabus**

Signals and systems –basic operations on signals – continuous time and discrete time signals – Continuous time and discrete time systems –properties of systems - Z-transform – region of convergence – properties of Z-transform – inverse Z-transform. Fourier transform (FT) of discrete time signals – properties of FT – relation between Z-transform and FT. Discrete Fourier transform (DFT) - Properties of DFT – inverse DFT - Fast Fourier transform (FFT) - Radix-2 FFT algorithms – butterfly structure. Digital filter structures –structures for IIR - Structures for FIR.

**Expected Outcome**

Student is able to

1. *Identify different types of continuous time and discrete time signals.*
2. *Identify different types of continuous time and discrete time systems.*
3. *Analyse signals using Z Transform and FT.*
4. *Analyse signals using DFT and FFT.*
5. *Appreciate IIR digital filter structures.*
6. *Appreciate FIR digital filter structures.*

**Text Books**

1. M.N. Bandyopadhyaya , Introduction to Signals and Systems and Digital Signal Processing, PHI.
2. S.D. Apte, Digital Signal Processing , Wiley India.

**References**


1. Li Tan , Digital Signal Processing, Fundamentals and Applications, Elsevier.
2. M. H. Hayes, Digital Signal Processing, Tata McGrawHill (SCHAUM'S OUTlines).
3. A.V. Oppenheim and R. W. Schafer, Digital Signal Processing, Prentice-Hall Inc.
4. A. Ambardar, Digital Signal Processing: A Modern Introduction, Thomson India Edition.
5. J.K. Proakis and D.G. Manolakis, Introduction to Digital Signal Processing, MacMillan
6. S.K. Mitra, Digital Signal Processing, Wiley.
7. S.W. Smith, Digital Signal Processing : A Practical Guide for Engineers and Scientists, Elsevier India.
8. P. Ramesh Babu, Digital Signal Processing, Scitech Publications.
9. D. Ganesh Rao and V. P. Gejji, Digital Signal Processing Theory and Lab Practice, Sanguine Publishers.


**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| **I** | Signals and systems – introduction – basic operations on signals – continuous time and discrete time signals –step, impulse, ramp, exponential and sinusoidal functions. | **07** | **15 %** |
| **II** | Continuous time and discrete time systems –properties of systems – linearity, causality, time invariance, memory, stability, invertibility. Linear time invariant systems – convolution. | **07** | **15 %** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Z-transform – region of convergence – properties of Z-transform – inverse Z-transform. Fourier transform (FT) of discrete time signals – properties of FT – relation between Z-transform and FT. | **07** | **15 %** |
| **IV** | Discrete Fourier transform (DFT) - Properties of DFT – inverse DFT - Fast Fourier transform (FFT) - Radix-2 FFT algorithms – butterfly structure. | **07** | **15 %** |
| **SECOND INTERNAL EXAM** | | | |

| | | | |
|---|---|---|---|
| **V** | Digital filter structures – block diagram and signal flow graph representation – structures for IIR – direct form structure – Cascade form structure – parallel form structure – lattice structure. | **07** | **20 %** |
| **VI** | Structures for FIR – direct form structures – direct form structure of linear phase system – cascade form structure – frequency sampling structure – lattice structure. | **07** | **20 %** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II;All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules I and II; T*wo* questions have to be answered. Each question can have a maximum of three subparts
4. Part C
    a. Total marks : 12
    b. *Four*questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules III and IV;T*wo*questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40

b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS365 | OPTIMIZATION TECHNIQUES | 3-0-0-3 | 2015 |

**Course Objectives**

1. *To build an understanding on the basics of optimization techniques.*
2. *To introduce basics of linear programming and meta- heuristic search techniques.*

**Syllabus**

Basics of Operations Research - Formulation of optimization problems - Linear Programming - Transportation Problem - Assignment Problem - Network flow Problem - Tabu Search - Genetic Algorithm - Simulated Annealing – Applications.

**Expected Outcome**

Student is able to

1. *Formulate mathematical models for optimization problems.*
2. *Analyze the complexity of solutions to an optimization problem.*
3. *Design programs using meta-heuristic search concepts to solve optimization problems.*
4. *Develop hybrid models to solve an optimization problem.*

**Text Books**

1. Rao S.S., Optimization Theory and Applications, Wiley Eastern.
2. Hamdy A. Taha, Operations Research – An introduction, Prentice – Hall India.
3. G. Zapfel, R. Barune and M. Bogl, Meta heuristic search concepts: A tutorial with applications to production and logistics, Springer.

**References**

1. Gass S. I., Introduction to Linear Programming, Tata McGraw Hill.
2. Reeves C., Modern heuristic techniques for combinatorial problems, Orient Longman.
3. Goldberg, Genetic algorithms in Search, optimization and Machine Learning, Addison Wesley.
4. K. Deb, Optimization for engineering design – algorithms and examples, Prentice Hall of India.

| COURSE PLAN | | | |
|---|---|---|---|
| **Module** | **Contents** | **Hours** | **Sem. Exam Marks %** |
| **I** | Decision-making procedure under certainty and under uncertainty - Operations Research-Probability and decision- making- Queuing or Waiting line theory-Simulation and Monte- Carlo Technique-Nature and organization of optimization problems-Scope and hierarchy of optimization- Typical applications of optimization. | **08** | **15%** |
| **II** | Essential features of optimization problems - Objective function- Continuous functions - Discrete functions - Unimodal functions - Convex and concave functions, Investment costs and operating costs in objective function - Optimizing profitably constraints-Internal and external constraints-Formulation of optimization problems. Continuous functions - Discrete functions - Unimodal functions - Convex and concave functions. | **07** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Necessary and sufficient conditions for optimum of unconstrained functions-Numerical methods for unconstrained functions - One-dimensional search - Gradient-free search with fixed step size. Linear Programming - Basic concepts of linear programming - Graphical interpretation-Simplex method - Apparent difficulties in the Simplex method. | **06** | **15%** |
| **IV** | Transportation Problem, Loops in transportation table, Methods of finding initial basic feasible solution, Tests for optimality. Assignment Problem, Mathematical form of assignment problem, methods | **06** | **15%** |

| | | | |
|---|---|---|---|
| | of solution. | | |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Network analysis by linear programming and shortest route, maximal flow problem. Introduction to Non-traditional optimization, Computational Complexity – NP-Hard, NP-Complete. Tabu Search- Basic Tabu search, Neighborhood, Candidate list, Short term and Long term memory | **07** | **20%** |
| **VI** | Genetic Algorithms- Basic concepts, Encoding, Selection, Crossover, Mutation. Simulated Annealing - Acceptance probability, Cooling, Neighborhoods, Cost function. Application of GA and Simulated Annealing in solving sequencing and scheduling problems and Travelling salesman problem. | **08** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.

5. Part D
    a. Total marks : 18
    b. *Three*questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| **CS367** | **Logic for Computer Science** | **3-0-0-3** | **2015** |

**Pre-requisites**

1. **BE101-05** Introduction to Computing and Problem Solving
2. **CS205** Data Structures

**Course Objectives**

1. *To introduce the concepts of mathematical logic and its importance.*
2. *To discuss propositional, predicate, temporal and modal logic and their applications.*

**Syllabus**

Propositional Logic, Resolution, binary decision diagrams, Predicate logic, resolution, temporal logic, deduction, program verification, modal logic.

**Expected Outcome**

Student is able to

1. *Explain the concept of logic and its importance.*
2. *Understand fundamental concepts in propositional logic and apply resolution techniques.*
3. *Understand fundamental concepts in predicate logic and apply resolution techniques.*
4. *Understand fundamental concepts in temporal logic and apply resolution techniques.*
5. *Understand the concept of program verification and apply it in real-world scenarios.*
6. *Understand fundamental concepts in modal logic.*

**Text Books**

1. Modechai Ben-Ari, Mathematical Logic for Computer Science, Springer, 3/e, 2102.
2. Arindhama Singh, Logics for Computer Science, Prentice Hall India, 2004.

**Reference**

1. Michael Huth, Mark Ryan, Logic in Computer Science: Modeling and Reasoning about Systems, Cambridge University Press, 2005.

| Course Plan | | | |
|---|---|---|---|
| **Module** | Contents | **Hours** | **Sem. Exam Marks %** |
| **I** | Introductory Concepts: Mathematical Logic, Propositional Logic, First Order Logic, Modal and Temporal logic, Program Verification. (**Reading: Ben-Ari, Chapter 1**) Propositional Logic: Formulae and interpretations, Equivalence, Satisfiability& Validity, Semantic Tableaux, Soundness and Completeness. (**Reading: Ben-Ari, Chapter 2 except 2.4, Additional Reading : Singh, Chapter 1**) | **06** | **15%** |
| **II** | The Hilbert Deductive System, Derived Rules, Theorems and operators, Soundness and Completeness, Consistency. (**Reading: Ben-Ari, Chapter 3 except 3.7 and 3.8, Additional Reading : Singh, Chapter 1**) Resolution in Propositional Logic: Conjunctive Normal form, Clausal form, resolution rule. (**Reading: Ben-Ari, Chapter 4.1, 4,2, 4.3, Additional Reading : Singh, Chapter 1**) | **06** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Binary Decision Diagrams: Definition, Reduced and ordered BDD, Operators. **(Reading: Ben-Ari, Chapter 5.1 – 5.5**) Predicate Logic: Relations, predicates, formulae and interpretation, logical equivalence, semantic tableaux, soundness. **Reading: Ben-Ari, Chapter 7.1-7.6, Additional Reading : Singh, Chapter 2**) | **07** | **15%** |
| **IV** | The Hilbert deduction system for predicate logic.Functions, PCNF and clausal form, Herbrand | **08** | **15%** |

| | | | |
|---|---|---|---|
| | model.Resolution in predicate logic: ground resolution, substitution, unification, general resolution.<br>**Reading: Ben-Ari, Chapter 8.1-8.4, 9.1, 9.3, 10.1-10.4, Additional Reading : Singh, Chapter 2, Chapter 3**) | | |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Temporal logic: Syntax and semantics, models of time, linear time temporal logic, semantic tableaux.<br>Deduction system of temporal logic.<br>(**Reading: Ben-Ari, Chapter 13.1-13.5, 14.1-14.2**) | **07** | **20%** |
| **VI** | Program Verification: Need for verification, Framework for verification, Verification of sequential programs, deductive system, verification, synthesis.<br>**(Reading: Ben-Ari, Chapter 15.1-15.4, Additional Reading : Singh, Chapter 5**)<br>**Modal Logic:** Need for modal logic, Case Study: Syntax and Semantics of K, Axiomatic System KC,<br>**(Reading: Singh, Chapter 6.1-6.3**) | **08** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C

a. Total marks : 12
b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.

5. Part D
   a. Total marks : 18
   b. *Three* questionseach having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| **CS369** | Digital System Testing & Testable Design | 3-0-0-3 | 2015 |

**Pre-requisites**

1. **CS203** Switching Theory and Logic Design
2. **CS234** Digital Systems Lab

**Course Objectives**

1. T*o expose the students to the basics of digital testing techniques applied to VLSI circuits.*
2. *To introduce the concepts of algorithm development for automatic test pattern generation for digital circuits.*
3. *To discuss fundamentals of design for testability.*

**Syllabus**

Basic terminology used in testing - functional and structural models of digital systems -logic simulation for design verification and testing-fault modeling - fault simulation - testing for faults - design for testability.

**Expected Outcome**

Student is able to

1. *Appreciate the basics of VLSI testing and functions modeling of circuits.*
2. *Apply fault modeling using single stuck & multiple stuck modeling for combinational circuits.*
3. *Evaluate different methods for logic and fault simulations.*
4. *Generate test patterns using automatic test pattern generation methods like D, PODEM & FAN algorithms for combinational circuits.*
5. *Explain automatic test pattern generation using time frame expansion and simulation based method for sequential circuits.*
6. *Design digital circuits using scan path and self tests.*

**Text Books**

1. MironAbramovici, Melvin A. Breuer, Arthur D. Friedman, Digital Systems Testing and Testable Design, Jaico Publishers.
2. Michael L. Bushnell and Vishwani D. Agrawal, Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits, Springer.
3. Alexander Miczo, Digital Logic Testing and Simulation, Wiley.

**Reference**

1. ZainalabedinNavabi, Digital System test and testable design, Springer.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | **Fundamentals of Testing:** Testing & Diagnosis, testing at different levels of abstraction, errors & faults, modeling & evaluation,types of testing, test generation **Modeling:** Functional modeling at logic level, functional modeling at register level & structural models. | 06 | 15% |
| II | **Fault Modeling :** Logic fault models, Fault detection and redundancy, Fault equivalence & fault location, fault dominance, single stuck faults, multiple stuck fault models . | 06 | 15% |
| | **FIRST INTERNAL EXAM** | | |
| III | **Logic & fault Simulation:** Simulationfor verification& test evaluation, types of simulation – compiled code & Event driven, serial fault simulation, statistical method for fault simulation. | 07 | 15% |
| IV | **Combinational circuit test generation**: ATG for SSFs in combinational circuits – fault oriented ATG- fault independent ATG- random test generation, Sensitized path, D-algorithm, PODEM and FAN. | 07 | 15% |
| | **SECOND INTERNAL EXAM** | | |
| V | **Sequential circuit test generation:** ATPG for single clock synchronous circuits, time frame expansion method, simulation based sequential circuit ATPG – genetic algorithm. | 07 | 20% |
| VI | **Design for Testability:** introduction to testability, design for testability techniques, controllability and | 09 | 20% |

| | | | |
|---|---|---|---|
| | observability by means of scan registers, generic scan based designs – scan path, boundary scan, Introduction to BIST. | | |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern:**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four*questions each having 3 marks, uniformly covering modules I and II;All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three sub-parts
4. Part C
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three* questionseach having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical/design questions.

# S5 – LABORATORY COURSES

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS331 | SYSTEM SOFTWARE LAB | 0-0-3-1 | 2015 |

**Course Objectives**

*To build an understanding on design and implementation of different types of systemsoftware.*

**List of Exercises/Experiments: (Exercises/experiments marked with * are mandatory from each part. Total 12 Exercises/experiments are mandatory)**

*Part A*

1. Simulate the following non-preemptive CPU scheduling algorithms to find turnaround time and waiting time.
   a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority

2. Simulate the following file allocation strategies.
   a) Sequential b) Indexed c) Linked

3. Implement the different paging techniques of memory management.

4. Simulate the following file organization techniques *
   a) Single level directory b) Two level directory c) Hierarchical

5. Implement the banker's algorithm for deadlock avoidance.*

6. Simulate the following disk scheduling algorithms. *
   a) FCFS b)SCAN c) C-SCAN

7. Simulate the following page replacement algorithms
   a) FIFOb)LRU c) LFU

8. Implement the producer-consumer problem using semaphores. *

9. Write a program to simulate the working of the dining philosopher's problem.*

*Part B*

10. Implement the symbol table functions: create, insert, modify, search, and display.
11. Implement pass one of a two pass assembler. *
12. Implement pass two of a two pass assembler. *
13. Implement a single pass assembler. *
14. Implement a two pass macro processor *
15. Implement a single pass macro processor.
16. Implement an absolute loader.
17. Implement a relocating loader.
18. Implement pass one of a direct-linking loader.
19. Implement pass two of a direct-linking loader.
20. Implement a simple text editor with features like insertion / deletion of a character, word, and sentence.
21. Implement a symbol table with suitable hashing.*

**Expected Outcome**

Student is able to

1. *Compare and analyze CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority.*
2. *Implement basic memory management schemes like paging.*
3. *Implement synchronization techniques using semaphores etc.*
4. *Implement banker's algorithm for deadlock avoidance.*
5. *Implement memory management schemes and page replacement schemes and file allocation and organization techniques.*
6. *Implement system software such as loaders, assemblers and macro processor.*

| Course No. | Course Name | L-T-P - Credits | Year of Introduction |
|---|---|---|---|
| CS333 | APPLICATION SOFTWARE DEVELOPMENT LAB | 0-0-3-1 | 2015 |

**Pre-requisite**

1. **CS208** Principles of Database Design

**Course Objectives**

1. *To introduce basic commands and operations on database.*
2. *To introduce stored programming concepts (PL-SQL)using Cursors and Triggers .*
3. *To familiarize front end tools of database.*

**List of Exercises/Experiments: (Exercises/experiments marked with * are mandatory from each part. Total 12 Exercises/experiments are mandatory)**

1. Creation of a database using DDL commands and writes DQL queries to retrieve information from the database.
2. Performing DML commands like Insertion, Deletion, Modifying, Altering, and Updating records based on conditions.
3. Creating relationship between the databases. *
4. Creating a database to set various constraints. *
5. Practice of SQL TCL commands like Rollback, Commit, Savepoint.
6. Practice of SQL DCL commands for granting and revoking user privileges.
7. Creation of Views and Assertions *
8. Implementation of Build in functions in RDBMS *
9. Implementation of various aggregate functions in SQL *
10. Implementation of Order By, Group By& Having clause. *
11. Implementation of set operators, nested queries and Join queries *
12. Implementation of various control structures using PL/SQL *
13. Creation of Procedures and Functions *
14. Creation of Packages *
15. Creation of database Triggers and Cursors *
16. Practice various front-end tools and report generation.
17. Creating Forms and Menus
18. Mini project (Application Development using Oracle/ MySQL using Database connectivity)*

a. Inventory Control System.
b. Material Requirement Processing.
c. Hospital Management System.
d. Railway Reservation System.
e. Personal Information System.
f. Web Based User Identification System.
g. Timetable Management System.
h. Hotel Management System.

**Expected Outcome**

Student is able to

1. *Design and implement a database for a given proble\ \ \ \m using database design principles.*
2. *Apply stored programming concepts (PL-SQL) using Cursors and Triggers.*
3. *Use graphical user interface, Event Handling and Database connectivity to develop and deploy applications and applets.*
4. *Develop medium-sized project in a team.*

# B-Tech Computer Science & Engineering

## Sixth Semester

## Course Syllabus & Course Plan

# S6 - CORE COURSES

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| CS302 | Design and Analysis of Algorithms | 3-1-0-4 | 2015 |

**Course Objectives**

1. *To introduce the concepts of Algorithm Analysis, Time Complexity, Space Complexity.*
2. *To discuss various Algorithm Design Strategies with proper illustrative examples.*
3. *To introduce Complexity Theory.*

**Syllabus**

Introduction to Algorithm Analysis, Notions of Time and Space Complexity, Asymptotic Notations, Recurrence Equations and their solutions, Master's Theorem, Divide and Conquer and illustrative examples, AVL trees, Red-Black Trees, Union-find algorithms, Graph algorithms, Divide and Conquer, Dynamic Programming, Greedy Strategy, Back Tracking and Branch and Bound, Complexity classes

**Expected outcome**

Student is able to

1. *Analyze a given algorithm and express its iime and space complexities in asymptotic notations.*
2. *Solve recurrence equations using Iteration Method, Recurrence Tree Method and Master's Theorem.*
3. *Design algorithms using Divide and Conquer Strategy.*
4. *Compare Dynamic Programming and Divide and Conquer Strategies.*
5. *Solve Optimization problems using Greedy strategy.*
6. *Design efficient algorithms using Back Tracking and Branch Bound Techniques for solving problems.*
7. *Classify computational problems into P, NP, NP-Hard and NP-Complete.*

**Text Books**

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, MIT Press [Modules 1,2,6]
2. Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, Computer Algorithms, Universities Press, 2007 [Modules 3,4,5]

**References**


1. AnanyLevitin, Introduction to the Design and Analysis of Algorithms, Pearson, 3rd Edition.
2. Richard E. Neapolitan,KumarssNaimipour, Foundations of Algorithms using C++ Psuedocode, Second Edition.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms, Pearson Education, 1999.
4. Gilles Brassard, Paul Bratley, Fundamentals of Algorithmics, Pearson Education.


**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | ***Introduction to Algorithm Analysis***Time and Space Complexity- Elementary operations and Computation of Time Complexity- Best, worst and Average Case Complexities- Complexity Calculation of simple algorithms<br>***Recurrence Equations:***Solution of Recurrence Equations – Iteration Method and Recursion Tree Methods, | **04**<br>**04** | **15 %** |
| II | ***Master's Theorem***(Proof not required) – examples, Asymptotic Notations and their properties- Application of Asymptotic Notations in Algorithm Analysis- Common Complexity Functions<br>***AVL Trees*** – rotations, Red-Black Trees insertion and deletion (Techniques only; algorithms not expected). B-Trees – insertion and deletion operations. Sets- Union and find operations on disjoint sets. | **04**<br>**05** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| III | ***Graphs*** – DFS and BFS traversals, complexity, Spanning trees – Minimum Cost Spanning Trees, single source shortest path algorithms, Topological sorting, strongly connected components. | **07** | **15%** |

| | | | |
|---|---|---|---|
| **IV** | ***Divide and Conquer:***The Control Abstraction, 2 way Merge sort, Strassen's Matrix Multiplication, Analysis<br>***Dynamic Programming :*** The control Abstraction- The Optimality Principle- Optimal matrix multiplication, Bellman-Ford Algorithm | **04**<br>**04** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Analysis, Comparison of Divide and Conquer and Dynamic Programming strategies<br>***Greedy Strategy: -*** The Control Abstraction- the Fractional Knapsack Problem,<br>Minimal Cost Spanning Tree Computation- Prim's Algorithm – Kruskal's Algorithm. | **02**<br>**04**<br>**03** | **20%** |
| **VI** | ***Back Tracking: -***The Control Abstraction – The N Queen's Problem, 0/1 Knapsack Problem<br>***Branch and Bound:***Travelling Salesman Problem.<br>***Introduction to Complexity Theory :-***Tractable and Intractable Problems- The P and NP Classes- Polynomial Time Reductions - The NP- Hard and NP-Complete Classes | **03**<br>**03**<br>**03** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18

b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.

4. Part C
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS304 | COMPILER DESIGN | 3-0-0-3 | 2015 |

**Course Objectives**

*To provide a thorough understanding of the internals of Compiler Design.*

**Syllabus**

Phases of compilation, Lexical analysis, Token Recognition, Syntax analysis, Bottom Up and Top Down Parsers, Syntax directed translation schemes, Intermediate Code Generation, Triples and Quadruples, Code Optimization, Code Generation.

**Expected Outcome**

Student is able to

1. *Explain the concepts and different phases of compilation with compile time error handling.*
2. *Represent language tokens using regular expressions, context free grammar and finite automata and design lexical analyzer for a language.*
3. *Compare top down with bottom up parsers, and develop appropriate parser to produce parse tree representation of the input.*
4. *Generate intermediate code for statements in high level language.*
5. *Design syntax directed translation schemes for a given context free grammar.*
6. *Apply optimization techniques to intermediate code and generate machine code for high level language program.*

**Text Books**

1. Aho A. Ravi Sethi and D Ullman. Compilers – Principles Techniques and Tools, Addison Wesley.
2. D. M.Dhamdhare, System Programming and Operating Systems,Tata McGraw Hill & Company.

**References**

1. Kenneth C. Louden, Compiler Construction – Principles and Practice,Cengage Learning Indian Edition.
2. Tremblay and Sorenson,The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company.

| Course Plan | | | |
|---|---|---|---|
| **Module** | **Contents** | **Hours** | **Sem. Exam Marks %** |
| **I** | Introduction to compilers – Analysis of the source program, Phases of a compiler, Grouping of phases, compiler writing tools – bootstrapping<br>**Lexical Analysis**:<br>The role of Lexical Analyzer, Input Buffering, Specification of Tokens using Regular Expressions, Review of Finite Automata, Recognition of Tokens. | **07** | **15%** |
| **II** | **Syntax Analysis:**<br>Review of Context-Free Grammars – Derivation trees and Parse Trees, Ambiguity.<br>**Top-Down Parsing**: Recursive Descent parsing, Predictive parsing, LL(1) Grammars. | **06** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | **Bottom-Up Parsing**:<br>Shift Reduce parsing – Operator precedence parsing (Concepts only)<br>LR parsing – Constructing SLR parsing tables, Constructing, Canonical LR parsing tables and Constructing LALR parsing tables. | **07** | **15%** |
| **IV** | **Syntax directed translation**:<br>Syntax directed definitions, Bottom- up evaluation of S-attributed definitions, L- attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes.<br>**Type Checking :**<br>Type systems, Specification of a simple type checker. | **08** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | **Run-Time Environments:**<br>Source Language issues, Storage organization, Storage- | | |

| | | | |
|---|---|---|---|
| | allocation strategies.<br>**Intermediate Code Generation (ICG)**:<br>Intermediate languages – Graphical representations, Three-Address code, Quadruples, Triples. Assignment statements, Boolean expressions. | **07** | **20%** |
| **VI** | **Code Optimization:**Principal sources of optimization, Optimization of Basic blocks<br>**Code generation**:<br>Issues in the design of a code generator. The target machine, A simple code generator. | **07** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questionseach having *9* marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18

b. *Three* questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts

6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P - Credits | Year of Introduction |
|---|---|---|---|
| CS306 | Computer Networks | 3-0-0-3 | 2015 |

**Course Objectives**

1. *To build an understanding of the fundamental concepts of computer networking.*
2. *To introduce the basic taxonomy and terminology of computer networking.*
3. *To introduce advanced networking concepts.*

**Syllabus**

Concept of layering, LAN technologies (Ethernet), Flow and error control techniques, switching, IPv4/IPv6, routers and routing algorithms (distance vector, link state), TCP/UDP and sockets, congestion control, Application layer protocols.

**Expected Outcome**

Student is able to

1. *Describe the different aspects of networks, protocols and network design models.*
2. *Examine various Data Link layer design issues and Data Link protocols.*
3. *Analyse and compare different LAN protocols.*
4. *Compare and select appropriate routing algorithms for a network.*
5. *Examine the important aspects and functions of network layer, transport layer and application layer in internetworking.*

**Text Books**

1. Andrew S. Tanenbaum, Computer Networks, 4/e, PHI.
2. Larry L. Peterson & Bruce S. Dave, Computer Networks-A Systems Approach, 5/e, Morgan Kaufmann, 2011.
3. Behrouz A. Forouzan, Data Communications and Networking, 4/e, Tata McGraw Hill.

**References**

1. James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, 6/e.
2. Keshav, An Engineering Approach to Computer Networks, Addison Wesley, 1998.
3. William Stallings, Computer Networking with Internet Protocols, Prentice-Hall, 2004.
4. W. Richard Stevens. TCP/IP Illustrated volume 1, Addison-Wesley.
5. Fred Halsall, Computer Networking and the Internet, 5/e.
6. Request for Comments (RFC) Pages - IETF -https://www.ietf.org/rfc.html

| Course Plan | | | |
|---|---|---|---|
| **Module** | **Contents** | **Hours** | **Sem. Exam Marks** |
| **I** | Introduction – Uses – Network Hardware – LAN –MAN – WAN, Internetworks – Network Software – Protocol hierarchies – Design issues for the layers – Interface & Service – Service Primitives. Reference models – OSI – TCP/IP. | **07** | **15%** |
| **II** | Data Link layer Design Issues – Flow Control and ARQ techniques. Data link Protocols – HDLC. DLL in Internet. MAC Sub layer – IEEE 802 FOR LANs & MANs, IEEE 802.3, 802.4, 802.5. Bridges - Switches – High Speed LANs - Gigabit Ethernet. Wireless LANs - 802.11 a/b/g/n, 802.15.PPP | **08** | **15%** |
| **FIRST INTERNAL EXAMINATION** | | | |
| **III** | Network layer – Routing – Shortest path routing, Flooding, Distance Vector Routing, Link State Routing, RIP, OSPF, Routing for mobile hosts. | **07** | **15%** |
| **IV** | Congestion control algorithms – QoS. Internetworking – Network layer in internet. IPv4 - IP Addressing – Classless and Classfull Addressing. Sub-netting. | **07** | **15%** |
| **SECOND INTERNAL EXAMINATION** | | | |
| **V** | Internet Control Protocols – ICMP, ARP, RARP, BOOTP. Internet Multicasting – IGMP, Exterior Routing Protocols – BGP. IPv6 – Addressing – Issues, ICMPv6. | **07** | **20%** |
| **VI** | Transport Layer – TCP & UDP. Application layer –FTP, DNS, Electronic mail, MIME, SNMP. Introduction to World Wide Web. | **07** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E

2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II;All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS308 | Software Engineering and Project Management | 3-0-0-3 | 2015 |

**Pre-requisite:**

**Course Objectives**

1. *To introduce the fundamental concepts of software engineering.*
2. *To build an understanding on various phases of software development.*
3. *To introduce various software process models.*

**Syllabus**

Introduction to software engineering, Software process models, Software development phases, Requirement analysis, Planning, Design, Coding, Testing, Maintenance.

**Expected Outcome**

Student is able to

1. *Identify suitable life cycle models to be used.*
2. *Analyze a problem and identify and define the computing requirements to the problem.*
3. *Translate a requirement specification to a design using an appropriate software engineering methodology.*
4. *Formulate appropriate testing strategy for the given software system.*
5. *Develop software projects based on current technology, by managing resources economically and keeping ethical values.*

**References**

1. Roger S. Pressman, Software Engineering : A practitioner's approach, McGraw Hill publication, Eighth edition, 2014
2. Walker Royce, Software Project Management : A unified frame work, Pearson Education, 1998
3. Ian Sommerville, Software Engineering, University of Lancaster, Pearson Education, Seventh edition, 2004.

4. K. K.Aggarwal and Yogesh Singh, Software Engineering, New age International Publishers, Second edition, 2005.
5. S.A. Kelkar, Software Project Management: A concise study, PHI, Third edition, 2012.

**COURSE PLAN**

| Module | Contents | Hours | Sem. Exam Marks% |
|---|---|---|---|
| I | Introduction to software engineering- scope of software engineering – historical aspects, economic aspects, maintenance aspects, specification and design aspects, team programming aspects. Software engineering a layered technology – processes, methods and tools. Software process models – prototyping models, incremental models, spiral model, waterfall model. | 07 | 15% |
| II | Process Framework Models: Capability maturity model (CMM), ISO 9000. Phases in Software development – requirement analysis- requirements elicitation for software, analysis principles, software prototyping, specification. | 06 | 15% |
| **FIRST INTERNAL EXAM** | | | |
| III | Planning phase – project planning objective, software scope, empirical estimation models- COCOMO, single variable model, staffing and personal planning. Design phase – design process, principles, concepts, effective modular design, top down, bottom up strategies, stepwise | 07 | 15% |

| | | | |
|---|---|---|---|
| | refinement. | | |
| **IV** | Coding – programming practice, verification, size measures, complexity analysis, coding standards. Testing – fundamentals, white box testing, control structure testing, black box testing, basis path testing, code walk-throughs and inspection, testing strategies-Issues, Unit testing, integration testing, Validation testing, System testing. | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Maintenance-Overview of maintenance process, types of maintenance. Risk management: software risks - risk identification-risk monitoring and management. Project Management concept: People – Product-Process-Project. | **07** | **20%** |
| **VI** | Project scheduling and tracking: Basic concepts-relation between people and effort-defining task set for the software project-selecting software engineering task<br>Software configuration management: Basics and standards User interface design - rules. Computer aided software engineering tools - CASE building blocks, taxonomy of CASE tools, integrated CASE environment. | **08** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having *3* marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questions each having *9* marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12
   b. *Four* questions each having *3* marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three* questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

# S6 - ELECTIVES

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS362 | Computer Vision | 3-0-0-3 | 2015 |

**Pre-requisite**

**NIL**

**Course Objectives**

1. *To build an understanding on detailed models of image formation.*
2. *To expose the students to image feature detection and matching.*
3. *To introduce fundamental algorithms for pattern recognition.*
4. *To introduce various classification techniques.*
5. *To expose the students to various structural pattern recognition and feature extraction techniques.*

**Syllabus**

Image formation and Image model with Components of a vision system, Multiple images and the Geometry of multiple views, High level vision, Basics of pattern recognition, Linear discriminant based classifiers and tree classifiers, Unsupervised Methods, Recent Advances in Pattern Recognition.

**Expected Outcome**

Student is able to

1. *Appreciate the detailed models of image formation.*
2. *Analyse the techniques for image feature detection and matching.*
3. *Apply various algorithms for pattern recognition.*
4. *Examinevarious clustering algorithms.*
5. *Analyze structural pattern recognition and feature extraction techniques.*

**Text Books:**

1. David A. Forsyth & Jean Ponce, Computer vision – A Modern Approach, Prentice Hall, 2002.

2. Bernd Jahne and Horst HauBecker, Computer vision and Applications, Academic press, 2000.

**References**

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, 2004.
2. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, John Wiley, 2001.
3. S. Theodoridis and K. Koutroumbas, Pattern Recognition, 4th Ed., Academic Press, 2009.
4. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

**COURSE PLAN**

| Module | Contents | Hours | Sem. ExamMarks% |
|---|---|---|---|
| **I** | Image formation and Image model- Components of a vision system- Cameras- camera model and camera calibration- Radiometry- Light in space- Light in surface - Sources, shadows and shading. | **06** | **15%** |
| **II** | Multiple images-The Geometry of multiple views- Stereopsis- Affine structure from motion- Elements of Affine Geometry Affine structure and motion from two images- Affine structure and motion from multiple images- From Affine to Euclidean images. | **07** | **15%** |
| **FIRST INTERNAL EXAM** | | | |
| **III** | High level vision- Geometric methods- Model based vision- Obtaining hypothesis by pose consistency, pose clustering and using Invariants, Verification. | **07** | **15%** |

| | | | |
|---|---|---|---|
| **IV** | Introduction to pattern and classification, supervised and unsupervised learning, Clustering Vs classification, Bayesian Decision Theory- Minimum error rate classification Classifiers, discriminant functions, decision surfaces- The normal density and discriminant-functions for the Normal density. | **07** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | **Linear discriminant based classifiers and tree classifiers** Linear discriminant function based classifiers- Perceptron- Minimum Mean Squared Error (MME) method, Support Vector machine, Decision Trees: CART, ID3. | **07** | **20%** |
| **VI** | **Unsupervised Methods** Basics of Clustering; similarity / dissimilarity measures; clustering criteria. Different distance functions and similarity measures, K-means algorithm. **Recent Advances in Pattern Recognition** Neural network structures for pattern recognition, Pattern classification using Genetic Algorithms. | **08** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.

3. Part B
    a. Total marks : 18
    b. *Three*questions each having *9* marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
    a. Total marks : 12
    b. *Four* questions each having *3* marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three*questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| CS364 | Mobile Computing | 3-0-0-3 | 2015 |

**Pre-requisite**

1. **CS307** Data Communication

**Course Objectives**

1. *To impart basic understanding of the wireless communication systems.*
2. *To expose students to various aspects of mobile and ad-hoc networks.*

**Syllabus**

Mobile Computing Application and Services, Mobile Computing Architecture, Emerging Technologies, Intelligent Networks and Internet, Wireless LAN, Operating Systems, Security Issues in mobile computing.

**Expected Outcome**

Student is able to

1. *Explainvarious Mobile Computing application , services and architecture.*
2. *Understand various technology trends for next generation cellular wireless networks.*
3. *Describe protocol architecture of WLAN technology.*
4. *UnderstandSecurityIssues in mobile computing.*

**Text Books**

1. Asoke K. Talukder, Hasan Ahmad, Mobile Computing Technology- Application and Service Creation, 2nd Edition, McGraw Hill Education.
2. Jochen Schiller, Mobile Communications, Pearson Education Asia, 2008.
3. Theodore S. Rappaport, Wireless Communications Principles and Practice, 2/e, PHI, New Delhi, 2004.

**References**

1. Andrew S. Tanenbaum, Computer Networks, PHI, Third edition, 2003.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | Introduction to mobile computing, Middleware and Gateways, Application and services, Internet-Ubiquitous networks, Architecture and three-tier architecture for Mobile Computing, Design consideration for Mobile Computing. | 06 | 15% |

| | | | |
|---|---|---|---|
| **II** | Spread spectrum – Direct sequence, Frequency hoping. Medium Access Control - SDMA, FDMA, TDMA, CDMA, Cellular concepts- channel assignment strategy- hand off strategy interface and system capacity- improving coverage and capacity in cellular system, Satellite Systems-GEO, LEO, MEO. Wireless Communication Systems- Telecommunication Systems-GSM- GSM services & features, architecture -DECT features & characteristics, architecture. | **06** | **15%** |
| | **FIRST INTERNAL EXAM** | | |
| **III** | Wireless LANS: Wireless LAN Standards – IEEE 802 Protocol Architecture, IEEE 802.11 System Architecture, Protocol Architecture & Services,MAC Layer & Management. Mobile Ad Hoc and sensor Networks, BLUETOOTH: Architecture & Protocol Stack. | **07** | **15%** |
| **IV** | Intelligent Networks and Interworking- Call processing fundamentals, Intelligence in Networks.SS#7 signaling, In Conceptual Model, Softswitch, Technologies and interface for IN,SS7 Security, MAP Sec, Virtual Private Network, Design constraints in Application for Handheld Devices, Symbian OS architecture,application framework | **07** | **15%** |
| | **SECOND INTERNAL EXAM** | | |
| **V** | Mobile internet-mobile network layer-mobile IP-dynamic host configuration protocol-, mobile transport layer-implications of TCP on mobility-indirect TCP-snooping TCP- mobile TCP transmission-selective retransmission, Transaction oriented TCP- Support for mobility-file systems-WAP. | **07** | **20%** |
| **VI** | Security issues in mobile computing, Information Security, Components of Information Security, Next | **09** | **20%** |

| | | | |
|---|---|---|---|
| | Generation Networks-Converged scenario, Narrowband to broadband, OFDM,FAMA/DAMA,MPLS | | |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
    a. Total marks : 12
    b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
    a. Total marks : 18
    b. *Three*questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
    a. Total marks : 12
    b. *Four*questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
    a. Total marks : 18
    b. *Three* questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
    a. Total Marks: 40
    b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
    c. A question can have a maximum of three sub-parts.

| Course No. | Course Name | L-T-P Credits | Year of Introduction |
|---|---|---|---|
| CS366 | Natural language processing | 3-0-0-3 | 2015 |

**Course Objectives**

1. *To introduce the fundamentals of Language processing from the algorithmic viewpoint.*
2. *To discuss various issues those make natural language processing a hard task.*
3. *To discuss some applications of Natural Language Processing (NLP).*

**Syllabus**

Levels of Language Analysis, Syntax, Semantics and Pragmatics of Natural Language, Language Processing, Issues and approaches to solutions, Applications of Natural Language Processing (NLP).

**Expected Outcome**

Student is able to

1. *appreciate the fundamental concepts of Natural Language Processing.*
2. *design algorithms for NLP tasks.*
3. *develop useful systems for language processing and related tasks involving text processing.*

**Text Books**

1. James Allen, Natural Language Understanding, The Benjamin/Cummings Publishing Company Inc., Redwood City, CA.
2. D. Jurafsky and J. H. Martin, Speech and Language Processing, Prentice Hall India.

**References**

1. Charniak, Eugene, Introduction to Artificial intelligence, Addison-Wesley.
2. Ricardo Baeza-Yates and BerthierRibeiro-Neto, Modern Information Retrieval, Addison-Wesley,1999.
3. U. S. Tiwary and TanveerSiddiqui, Natural Language Processing and Information Retrieval, Oxford University Press.

**Course Plan**

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| I | Introduction to Natural Language Understanding- Levels of language analysis- Syntax, Semantics, Pragmatics.<br>Linguistic Background- An Outline of English Syntax. | 10 | 15% |
| II | Lexicons, POS Tagging, Word Senses.<br>Grammars and Parsing- Features, Agreement and | 10 | 15% |

| | | | |
|---|---|---|---|
| | Augmented Grammars. | | |
| **FIRST INTERNAL EXAM** | | | |
| **III** | Grammars for Natural Language, Parsing methods and Efficient Parsing.<br>Ambiguity Resolution- Statistical Methods. Probabilistic Context Free Grammar. | **12** | **15%** |
| **IV** | Semantics and Logical Form: Linking Syntax and Semantics- Ambiguity Resolution- other Strategies for Semantic Interpretation- Scoping and the Interpretation of Noun Phrases. | **08** | **15%** |
| **SECOND INTERNAL EXAM** | | | |
| **V** | Knowledge Representation and Reasoning- Local Discourse Context and Reference- Using World Knowledge- Discourse Structure- Defining a Conversational Agent. | **10** | **20%** |
| **VI** | Applications- Machine Translation, Information Retrieval and Extraction, Text Categorization and Summarization. | **06** | **20%** |
| **END SEMESTER EXAM** | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having *3* marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questions each having *9* marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12

b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All *four* questions have to be answered.

5. Part D
   a. Total marks : 18
   b. *Three* questions each having *9* marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts.
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| CS368 | Web Technologies | 3-0-0-3 | 2015 |

**Course Objectives**

1. *To impart the design, development and implementation of Dynamic Web Pages.*
2. *To develop programs for Web using Scripting Languages.*
3. *To expose the students to the world of XML.*

**Syllabus**

Basics of Internet and World Wide Web, HTML and XHTML, Cascading Style Sheets, Basics of JavaScript, Introduction to XML, Overview of PHP

**Expected Outcome**

Student is able to

1. *Develop interactive Web pages using HTML/XHTML.*
2. *Present a professional document using Cascaded Style Sheets.*
3. *Construct websites for user interactions using JavaScript.*
4. *Create XML documents for information interchange.*
5. *Develop Web applications using PHP.*
6. *Create and handle cookies for user identification.*

**Text Books**

1. Robert W.Sebesta, Programming the World Wide Web, 7/e, Pearson Education Inc., 2014.
2. P. J. Deitel, H.M. Deitel, Internet & World Wide Web How To Program, 4/e, Pearson International Edition 2010.

**References**

1. DreamTech, Web Technologies: HTML, JS, PHP, Java, JSP, ASP.NET, XML, AJAX, Black Book,Kogent Learning Solutions Inc. 2009.
2. Chris Bates, Web Programming Building Internet Applications, 3/e, Wiley India Edition 2009.
3. H.M.Deitel, P.J.Deitel, T.R. Nieto, T.M. Lin, P Sadhu, XML How To Program, Pearson Education Inc. 2011.
4. Anders Moller, Michael Schwartzbach, An Introduction to XML and Web Technologies, Pearson Education Inc. 2012.
5. Pankaj Sharma, Introduction to Web Technology, Katson Books, 2008.

6. Jeffrey C Jackson, Web Technologies A Computer Science Perspective, Pearson Education Inc. 2009.

Course Plan

| Module | Contents | Hours | Sem.ExamMarks % |
|---|---|---|---|
| **I** | **Introduction to the Internet:**The World Wide Web, Web Browsers, Web Servers, Uniform Resource Locators, Multipurpose Internet Mail Extensions, The Hypertext Transfer Protocol. | **06** | **15%** |
| **II** | **Introduction to HTML/XHTML :** Origins and Evolution of HTML and XHTML, Basic Syntax of HTML, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5, Syntactic Differences between HTML and XHTML. | **07** | **15%** |
| FIRST INTERNAL EXAM | | | |
| **III** | **Cascading Style Sheets :** Levels of Style Sheets - Style Specification Formats, Selector Forms, Property-Value Forms, Font Properties, List Properties, Alignment of Text, Color, The Box Model, Background Images, The span and div Tags, Conflict Resolution. | **06** | **15%** |
| **IV** | **The Basics of JavaScript :** Overview of JavaScript, Object Orientation and JavaScript, General Syntactic Characteristics- Primitives, Operations, and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Constructors, Pattern Matching Using Regular Expressions | **07** | **15%** |
| SECOND INTERNAL EXAMINATION | | | |
| **V** | **Introduction to XML:**The Syntax of XML,XML Document Structure, Namespaces, XML Schemas, Displaying Raw XML Documents, Displaying XML Documents with CSS, XSLT Style Sheets, XML Processors,Web Services. | **08** | **20%** |

| | | | |
|---|---|---|---|
| **VI** | **Introduction to PHP** :Origins and Uses of PHP, Overview of PHP - General Syntactic Characteristics - Primitives, Operations, and Expressions - Control Statements,Arrays, Functions, Pattern Matching, Form Handling, Cookies, Session Tracking. | **08** | **20%** |
| END SEMESTER EXAM | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40
   b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
   c. A question can have a maximum of three sub-parts.

| Course No. | Course Name | L-T-P -Credits | Year of Introduction |
|---|---|---|---|
| CS372 | HIGH PERFORMANCE COMPUTING | 3-0-0-3 | 2015 |

**Pre-requisites**

1. **BE101-05** Introduction to Computing & Problem Solving
2. **CS203** Switching Theory and Logic Design
3. **CS202** Computer Organization and Architecture

**Course Objectives**

1. *To introduce the concepts of Modern Processors.*
2. *To introduce Optimization techniques for serial code.*
3. *To introduce Parallel Computing Paradigms.*
4. *To introduce Parallel Programming using OpenMP and MPI.*

**Syllabus**

Modern processors - pipelining-superscalarity-multicore processors- Mutithreaded processors- vector processors- basic optimization techniques for serial code - taxonomy of parallel computing paradigms- shared memory computers- distributed-memory computers- Hierarchical Systems- networks- basics of parallelization - data parallelism - function parallelism- Parallel scalability- shared memory parallel programming with OpenMp - Distributed-memory parallel programming with MPI.

**Expected Outcome**

Student is able to

*1. appreciate the concepts used in Modern Processors for increasing the performance.*

*2. appreciate Optimization techniques for serial code.*

*3. appreciate Parallel Computing Paradigms.*

*4. identify the performance issues in Parallel Programming using OpenMP and MPI.*

**Text Book**

1. Georg Hager, Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall / CRC Computational Science series, 2011.

**References**

1. Charles Severance, Kevin Dowd, High Performance Computing, O'Reilly Media, 2nd Edition, 1998.
2. Kai Hwang, Faye Alaye Briggs, Computer Architecture and Parallel Processing, McGraw Hill, 1984.

Course Plan

| Module | Contents | Hours | Sem. Exam Marks % |
|---|---|---|---|
| **I** | Modern Processors : Stored Program Computer Architecture- General purpose cache- based microprocessor-Performance based metrics and benchmarks- Moore's Law- Pipelining- Superscalarity- SIMD- Memory Hierarchies Cache- mapping- prefetch- Multicore processors- Mutithreaded processors- Vector Processors- Design Principles- Maximum performance estimates- Programming for vector architecture. | **07** | **15%** |
| **II** | Basic optimization techniques for serial code : scalar profiling- function and line based runtime profiling- hardware performance counters- common sense optimizations- simple measures, large impact- elimination of common subexpressions- avoiding branches- using simd instruction sets- the role of compilers - general optimization options- inlining - aliasing- computational accuracy- register optimizations- using compiler logs- c++ optimizations - temporaries- dynamic memory management- loop kernels and iterators data access optimization: balance analysis and lightspeed estimates- storage order- case study: jacobi algorithm and dense matrix transpose. | **07** | **15%** |
| FIRST INTERNAL EXAM | | | |

| | | | |
|---|---|---|---|
| **III** | Parallel Computers : Taxonomy of parallel computing paradigms- Shared memory computers- Cache coherance- UMA - ccNUMA- Distributed-memory computers- Hierarchical systems- Networks- Basic performance characteristics- Buses- Switched and fat-tree networks- Mesh networks- Hybrids - Basics of parallelization - Why parallelize - Data Parallelism - Function Parallelism- Parallel Scalability- Factors that limit parallel execution- Scalability metrics- Simple scalability laws- parallel efficiency - serial performance Vs Strong scalability- Refined performance models- Choosing the right scaling baseline- Case Study: Can slow processors compute faster- Load balance. | **07** | **15%** |
| **IV** | Distributed memory parallel programming with MPI : message passing - introduction to MPI – example - messages and point-to-point communication - collective communication – nonblocking point-to-point communication- virtual topologies - MPI parallelization of Jacobi solver- MPI implementation - performance properties | **08** | **15%** |
| | SECOND INTERNAL EXAM | | |
| **V** | Shared memory parallel programming with OpenMp : introduction to OpenMp - parallel execution - data scoping- OpenMp work sharing for loops- synchronization - reductions - loop scheduling - tasking - case study: OpenMp- parallel jacobi algorithm- advanced OpenMpwavefront parallelization- Efficient OpenMP programming: Profiling OpenMP programs - Performance pitfalls- Case study: Parallel Sparse matrix-vector multiply. | **08** | **20%** |

| | | | |
|---|---|---|---|
| **VI** | Efficient MPI programming : MPI performance tools- communication parameters- Synchronization, serialization, contention- Reducing communication overhead- optimal domain decomposition- Aggregating messages – Nonblocking Vs Asynchronous communication- Collective communication- Understanding intra-node point-to-point communication. | **08** | **20%** |
| END SEMESTER EXAM | | | |

**Question Paper Pattern**

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules I and II; All*four* questions have to be answered.
3. Part B
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules I and II; *Two* questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
   a. Total marks : 12
   b. *Four* questions each having 3 marks, uniformly covering modules III and IV; All*four* questions have to be answered.
5. Part D
   a. Total marks : 18
   b. *Three* questions each having 9 marks, uniformly covering modules III and IV; *Two* questions have to be answered. Each question can have a maximum of three subparts
6. Part E
   a. Total Marks: 40

b. *Six* questions each carrying 10 marks, uniformly covering modules V and VI; *four* questions have to be answered.
c. A question can have a maximum of three sub-parts.

7. There should be at least 60% analytical/numerical questions.

# S6 – LABORATORY COURSES

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS332 | MICROPROCESSOR LAB | 0-0-3-1 | 2015 |

**Pre-requisite**

1. **CS305** Microprocessors and Microcontrollers

**Course Objectives**

1. *To practice assembly language programming on 8086.*
2. *To practice fundamentals of interfacing/programming various peripheral devices with microprocessor/microcontroller.*

**List of Exercises/ Experiments: (Minimum 12 Exercises/ Experiments are mandatory. Exercises/ Experiments marked with * are mandatory)**

**I. Assembly Language Programming Exercises/Experiments using 8086 Trainer kit**

1. Implementation of simple decimal arithmetic and bit manipulation operations.*
2. Implementation of code conversion between BCD, Binary, Hexadecimal and ASCII.
3. Implementation of searching and sorting of 16-bit numbers.
4. Programming exercises using stack and subroutines.*

**II. Exercises/Experiments using MASM (PC Required)**

5. Study of Assembler and Debugging commands.
6. Implementation of decimal arithmetic( 16 and 32 bit) operations.*
7. Implementation of String manipulations.*
8. Implementation of searching and sorting of 16-bit numbers.
9. Implementation of Matrix operations like addition, transpose, multiplication etc.

**III. Interfacing Exercises/Experiments with 8086 trainer kit through Assembly Language Programming**

10. Interfacing with stepper motor - Rotate through any given sequence.*
11. Interfacing with 8255 (mode0 and mode1 only).*
12. Interfacing with 8279 (Rolling message, 2 key lock out and N-key roll over implementation).*
13. Interfacing with 8253/54 Timer/Counter.

14. Interfacing with Digital-to-Analog Converter.*
15. Interfacing with Analog-to- Digital Converter.
16. Interfacing with 8259 Interrupt Controller.

**IV. Exercises/Experiments using 8051 trainer kit**

17. Familiarization of 8051 trainer kit by executing simple Assembly Language programs such as decimal arithmetic and bit manipulation.*
18. Implementation of Timer programming (in mode1).
19. Implementation of stepper motor interfacing, ADC/DAC interfacing and sensor interfacing with 8251 through Assembly Language programming.

**Expected Outcome**

Student is able to

1. *Develop assembly language programs for problem solving using software interrupts and various assembler directives.*
2. *Implementinterfacing of various I/O devices to the microprocessor/microcontroller through assembly language programming.*

| Course No. | Course Name | L-T-P-Credits | Year of Introduction |
|---|---|---|---|
| CS334 | Network Programming Lab | 0-0-3-1 | 2015 |

**Pre-requisite**

1. **CS307** Data Communication
2. **CS306** Computer Networks

**Course Objectives**

1. *To introduce Network related commands and configuration files in Linux Operating System.*
2. *To introduce tools for Network Traffic Analysis and Network Monitoring.*
3. *To practice Network Programming using Linux System Calls.*
4. *To design and deploy Computer Networks.*

**List of Exercises/ Experiments (12 Exercises/ Experiments are to be completed . Exercises/ Experiments marked with * are mandatory)**

1. Getting started with Basics of Network configurations files and Networking Commands in Linux.
2. To familiarize and understand the use and functioning of System Calls used for Operating system and network programming in Linux.
3. Familiarization and implementation of programs related to Process and thread.
4. Implement banker's algorithm for deadlock avoidance.
5. Implement Dining philosopher's problem for process synchronization.
6. Implement programs for Inter Process Communication using PIPE, Message Queue and Shared Memory.
7. Implement Client-Server communication using Socket Programming and TCP as transport layer protocol.*
8. Implement Client-Server communication using Socket Programming and UDP as transport layer protocol.*
9. Implement a multi user chat server using TCP as transport layer protocol.*
10. Implement Concurrent Time Server application using UDP to execute the program at remoteserver. Client sends a time request to the server, server sends its system time back to the client. Client displays the result.*
11. Implement and simulate algorithm for Distance vector routing protocol.
12. Implement and simulate algorithm for Link state routing protocol.

13. Implement Simple Mail Transfer Protocol.*
14. Develop concurrent file server which will provide the file requested by client if it exists. If not server sends appropriate message to the client. Server should also send its process ID (PID) to clients for display along with file or the message.*
15. Using Wireshark observe data transferred in client server communication using UDP and identify the UDP datagram.
16. Using Wireshark observe Three Way Handshaking Connection Establishment, Data Transfer and Three Way Handshaking Connection Termination in client server communication using TCP.
17. Develop a packet capturing and filtering application using raw sockets.
18. Design and configure a network with multiple subnets with wired and wireless LANs using required network devices. Configure the following services in the network- TELNET, SSH, FTP server, Web server, File server, DHCP server and DNS server.*
19. Install network simulator NS-2 in any of the Linux operating system and simulate wired and wireless scenarios.

**Expected Outcome**

Student is able to

1. *Usenetwork related commands and configuration files in Linux Operating System.*
2. *Developoperating system and network application programs.*
3. *Analyze network traffic using network monitoring tools.*