Indira Gandhi University, Meerpur, Rewari
SCHEME OF STUDIES AND EXAMINATION
B.TECH. (Computer Science & Engineering)
SEMESTER 5th AND 6th
Scheme effective from 2020-21

COURSE CODE AND DEFINITIONS:

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>Lecture</td>
</tr>
<tr>
<td>T</td>
<td>Tutorial</td>
</tr>
<tr>
<td>P</td>
<td>Practical</td>
</tr>
<tr>
<td>BSC</td>
<td>Basic Science Courses</td>
</tr>
<tr>
<td>ESC</td>
<td>Engineering Science Courses</td>
</tr>
<tr>
<td>HSMC</td>
<td>Humanities and Social Sciences including Management courses</td>
</tr>
<tr>
<td>PCC</td>
<td>Professional Core Courses</td>
</tr>
<tr>
<td>LC</td>
<td>Laboratory Courses</td>
</tr>
<tr>
<td>MC</td>
<td>Mandatory Courses</td>
</tr>
<tr>
<td>PT</td>
<td>Practical Training</td>
</tr>
<tr>
<td>S</td>
<td>Seminar</td>
</tr>
<tr>
<td>TH</td>
<td>Theory</td>
</tr>
<tr>
<td>Pr</td>
<td>Practical</td>
</tr>
</tbody>
</table>
General Notes:

1. Mandatory courses are non credit courses in which students will be required passing marks in internal assessments.

2. Students will be allowed to use non programmable scientific calculator. However, sharing of calculator will not be permitted in the examination.

3. Students will be permitted to opt for any elective course run by the department. However, the department shall offer those electives for which they have expertise. The choice of the students for any elective shall not be binding for the department to offer, if the department does not have expertise. To run the elective course a minimum of 1/3\textsuperscript{rd} students of the class should opt for it.
## Scheme of Studies and Examination
### B.TECH (Computer Science & Engineering) – 5th Semester
**w.e.f. 2020-21**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Category</th>
<th>Course Code</th>
<th>Course Title</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Total Contact Hrs. per week</th>
<th>Credit</th>
<th>Examination Schedule (Marks)</th>
<th>Duration of Exam (Hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Engineering Science Course</td>
<td>ESC-CSE-301</td>
<td>Microprocessor</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>2</td>
<td>Professional Core Course</td>
<td>PCC-CSE-303</td>
<td>Computer Networks</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>3</td>
<td>Professional Core Course</td>
<td>PCC-CSE-305</td>
<td>Formal Languages &amp; Automata</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>4</td>
<td>Professional Core Course</td>
<td>PCC-CSE-307</td>
<td>Design &amp; Analysis of Algorithms</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>5</td>
<td>Professional Core Course</td>
<td>PCC-CSE-309</td>
<td>Programming in Java</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>6</td>
<td>Professional Elective Course</td>
<td>Refer to Annexure I</td>
<td>Elective–I</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>75</td>
</tr>
<tr>
<td>7</td>
<td>Engineering Science Course</td>
<td>LC-ESC-321</td>
<td>Microprocessor Lab</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>25</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>LC-CSE-323</td>
<td>Computer Networks Lab</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>1.5</td>
<td>25</td>
<td>-</td>
</tr>
<tr>
<td>9</td>
<td>Professional Core Course</td>
<td>LC-CSE-325</td>
<td>Design &amp; Analysis of Algorithms Using C++</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>1.5</td>
<td>25</td>
<td>-</td>
</tr>
<tr>
<td>10</td>
<td>Professional Core Course</td>
<td>LC-CSE-327</td>
<td>Programming in Java Lab</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>1.5</td>
<td>25</td>
<td>-</td>
</tr>
<tr>
<td>11</td>
<td>Training</td>
<td>PT-CSE-329</td>
<td>Practical Training-1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**TOTAL CREDIT** 23.5 800

**Note:**

1. The evaluation of Practical Training-I will be based on seminar, viva-voce, report submitted by the students. According to performance, the students are awarded grades A, B, C, F. A student who is awarded ‘F’ grade is required to repeat Practical Training.

   **Excellent:** A; **Good:** B; **Satisfactory:** C; **Not Satisfactory:** F.
<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Category</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Total Contact Hrs. per week</th>
<th>Credit</th>
<th>Examination Schedule (Marks)</th>
<th>Duration of Exam (Hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Mark of Class work</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Theory</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Practical</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Total</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Professional Core Course</td>
<td>PCC-CSE-302</td>
<td>Compiler Design</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Professional Core Course</td>
<td>PCC-CSE-304</td>
<td>Artificial Intelligence</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Professional Core Course</td>
<td>PCC-CSE-306</td>
<td>Advanced Java</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Engineering Science Course</td>
<td>ESC-CSE-308</td>
<td>Data Science</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Professional Elective Course</td>
<td>Refer to Annexure II</td>
<td>Elective-II</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Professional Elective Course</td>
<td>Refer to Annexure III</td>
<td>Elective-III</td>
<td>3 0 0 3 3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>Project</td>
<td>PROJ-CSE-322</td>
<td>Project-I</td>
<td>0 0 4 4 2</td>
<td>25</td>
<td>25</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>LC-CSE-324</td>
<td>Compiler Design Lab</td>
<td>0 0 3 3 1.5</td>
<td>25</td>
<td>25</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>Professional Core Course</td>
<td>LC-CSE-326</td>
<td>Artificial Intelligence Lab using python</td>
<td>0 0 3 3 1.5</td>
<td>25</td>
<td>25</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>Professional Core Course</td>
<td>LC-CSE-328</td>
<td>Advanced Java Lab</td>
<td>0 0 2 2 1</td>
<td>25</td>
<td>25</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>Mandatory Courses</td>
<td>MC-317</td>
<td>Constitution of India</td>
<td>2 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** At the end of 6th semester each student has to undergo Practical Training of 4/6 weeks in an Industry/ Institute/ Professional Organization/ Research Laboratory/ training centre etc. and submit typed report along with a certificate from the organization & its evaluation shall be carried out in the 7th Semester.

'**MC-317**' is a mandatory non –credit course in which the students will be required passing marks in theory.
Annexure I

Elective –I (Professional Elective Course)

1. PEC-CSE-311: Software Engineering
2. PEC-CSE-313: System Programming and System Administration
3. PEC-CSE-315: Digital Image Processing

Annexure II

Elective –II (Professional Elective Course)

1. PEC-CSE-310: Advanced Database Management System
2. PEC-CSE-312: Mobile Application Development
3. PEC-CSE-314: Computer Graphics
4. PEC-CSE-330: Communication Engineering

Annexure III

Elective –III (Professional Elective Course)

1. PEC-CSE-316: Distributed System
2. PEC-CSE-318: Information Technology & Industry Business Skills
3. PEC-CSE-320: Mobile and Wireless Communication
MICROPROCESSOR

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>ESC-CSE-301</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Engineering Science Course</td>
</tr>
<tr>
<td>Coursedtitle</td>
<td>Microprocessor</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Course Objectives:
- To make understand architecture and working of Intel 8085 microprocessor in depth.
- To make understand architecture and working of Intel 8086 microprocessor in depth.
- Familiarization with the assembly language programming.
- Familiarization with various peripheral operations

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1
THE 8085 PROCESSOR: Introduction to microprocessor, 8085 microprocessor: Architecture, instruction set, interrupt structure, and Assembly language programming.

Unit: 2
THE 8086 MICROPROCESSOR ARCHITECTURE: Architecture, block diagram of 8086, details of sub-blocks such as EU, BIU; memory segmentation and physical address computations, program relocation, addressing modes, instruction formats, pin diagram and description of various signals.

Unit: 3
INSTRUCTION SET OF 8086: Instruction execution timing, assembler instruction format, data transfer instructions, arithmetic instructions, branch instructions, looping instructions, NOP and HLT instructions, flag manipulation instructions, logical instructions, shift and rotate instructions, directives and operators, programming examples.

Unit: 4
INTERFACING DEVICE: 8255 Programmable peripheral interface, interfacing keyboard and seven segment display, 8254 (8253) programmable interval timer, 8259A programmable interrupt controller, Direct Memory Access and 8237 DMA controller.

TEXT BOOKS:
1. Microprocessor Architecture, Programming & Applications with 8085: Ramesh S Gaonkar; Wiley Eastern Ltd.
2. Intel Microprocessors 8086- Pentium processor: Brey; PHI
REFERENCE BOOKS:
1. Microprocessors and interfacing: D V Hall; TMH
2. The 8088 & 8086 Microprocessors-Programming, interfacing, Hardware & Applications: Triebel & Singh; PHI
4. Advanced Microprocessors and Interfacing: Badri Ram; TMH

Course Outcomes:

- Understand the operation and architecture of Intel 8085 microprocessor including Instruction Set Architecture, assembly language programming, timing and speed of operation.
- Learn the operation of circuits for user interaction through switches, keyboard and display devices.
- Understand the operation and architecture of Intel 8086 microprocessor including Instruction Set Architecture, assembly language programming, timing and speed of operation.
- Understand the motivation and need for peripheral operations circuits for digital data exchange, timer, serial communication, merits of direct memory access, interrupt controller and other circuits.
COMPUTER NETWORKS

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PCC-CSE-303</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Computer Networks</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

**Course Objectives:**
- To develop an understanding of modern network architectures from a design and Performance perspective.
- To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
- To provide an opportunity to do Network programming
- To provide a WLAN measurement idea.

**Note:** Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

**Unit: 1**
**Introduction:** Data communication, Components, Computer networks and its historical development, distributed processing, Internet

**Network Models:** OSI model and TCP/IP Model

**Physical Layer** – physical layer functions, Data Representation, Simplex, Half Duplex and Full Duplex Transmission, Modulation and Multiplexing, Packet and circuit switching, Transmission media, Topologies, connectionless and connection-oriented services.

**Data Link Layer** : Data link layer functions and services, MAC Addressing, Framing, Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window Protocol.

**Unit: 2**
**Medium Access Control**: MAC layer functions, Random access, Controlled Access and channelization protocols.

**Network Layer**: Network layer functions and services, Logical addressing, IPv4 classful and classless addressing, subnetting, NAT, IPv4, ICMPv4, ARP, RARP and BOOTP, IPv6, IPv6 addressing, DHCP.

**Network Devices**: Repeater, hub, switch, router and gateway.
Unit: 3
Routing Algorithms: introduction to routing, Shortest Path Algorithm, Flooding, Hierarchical Routing, Link State and Distance Vector Routing

Transport Layer: Transport layer functions and services, Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), TCP connection management

Application Layer: Application layer functions and services, Domain Name Space (DNS), EMAIL, File Transfer Protocol (FTP), HTTP, SNMP

Unit: 4
Congestion Control, Quality of Service, QoS Improving techniques.
LAN: Ethernet, Token Bus, Token Ring, MAN Architecture- DQDB, WAN Architectures- Frame Relay, ATM, SONET/SDH

Network Security: Firewalls, security goals, types of attack, Introduction to cryptography, Types of ciphers: symmetric and asymmetric key ciphers.

Suggested books:

Suggested reference books:
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

Course Outcomes:
- Explain the functions of the different layer of the OSI Protocol.
- Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) and describe the function of each.
- Identify and connect various connecting components of a computer network.
- Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.
Course Objectives:
- To understand basic concepts of formal languages and automata theory.
- To study the types of Automata i.e. NFA, DFA, NFA with ε-transition and their inter conversion methods and importance.
- To Study formal languages of different kinds, such as regular and context-free languages. Understand the concept of grammar and its types. Removal of ambiguity and reduced form and Normal forms of grammar.
- To develop the concepts and design of higher-level automata to accept the language not accepted by finite automata such as PDA & Turing machine.
- To study the various properties of Turing machine and their designing.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit 1:

Finite Automata: Introduction: Set, Power Set, Super Set, Alphabet, languages and grammars, productions and derivation, Deterministic finite automata (DFA), Non-Deterministic finite automata (N DFA), Equivalence of DFA and NDFA, Conversion of NFA to DFA, minimization of finite automata, Finite automata with ε-moves, Acceptability of a string by a finite Automata.

Introduction to Machines: Properties and limitations of Finite Automata, Mealy and Moore Machines, Equivalence of Mealy and Moore machines.

Unit 2:

Regular Expression: State and prove Arden's Method, Regular Expressions, Recursive definition of regular expression, Regular expression conversion to Finite Automata and vice versa.

Properties of regular languages: Regular language, pumping lemma for regular sets/languages, Application of regular languages.
Unit 3:

**Grammars:** Chomsky hierarchy of languages, Relation between different types of grammars, Context-free grammar, Derivation tree / Parse tree, Ambiguity in regular grammar and their removal, Reduced Forms: Removal of useless symbols, null and unit productions, Normal Form: Chomsky Normal form(CNF) and Greibach Normal Form(GNF),

**Push Down Automata:** Introduction to PDA, Deterministic and Non-Deterministic PDA, Design of PDA: Transition table, Transition diagram and acceptability of strings by designed PDA, Pushdown automata (PDA) and equivalence with CFG.

Unit 4:

**Turing machines:** The basic model for Turing machines (TM), Deterministic and Non-Deterministic Turing machines and their equivalence, Design of Turing Machines: Transition table, Transition diagram and acceptability of strings by designed turing machine. Variants of Turing machines, Halting problem of Turing machine, PCP Problem of Turing Machine, Linear Bounded Automata, TMs as enumerators.

**Undecidability:** Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice’s theorem, undecidable problems about languages.

**Suggested books:**

**Suggested reference books**

**Course Outcomes:**
- To use basic concepts of formal languages of finite automata techniques.
- To Design Finite Automata’s for different Regular Expressions and Languages.
- To Construct context free grammar for various languages.
- To solve various problems of applying normal form techniques, push down automata and Turing Machines.
Course Objectives:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit 1:
**Introduction to Algorithms:** Algorithm, Performance Analysis (Time and Space complexity), Asymptotic Notation (Big OH, Omega and Theta)-best, average and worst-case behaviour. Elementary Data Structures (Basic terminology of Stacks and Queues, Tree, Graph), Sets and Disjoint Set Union.

**Divide and Conquer:** General method, Binary Search, Merge Sort, Quick Sort, and other sorting algorithms with divide and conquer strategy, Strassen’s Matrix Multiplication algorithms and analysis of these problems.

Unit 2:
**Greedy Method:** General method, Fractional Knapsack problem, Job Sequencing with Deadlines, Minimum Cost Spanning Trees, Single source shortest paths.

**Dynamic Programming:** General method, Optimal Binary Search Trees, 0/1 knapsack, The Traveling Salesperson problem.

Unit 3:
**Back Tracking:** General method, The 8-Queen’s problem, Sum of subsets, Graph Colouring, Hamiltonian Cycles.

**Branch and Bound:** The method, 0/1 knapsack problem, Traveling Salesperson problem, Efficiency considerations.
Unit 4:


Suggested Text Books:
1. Fundamental of Computer algorithms, Ellis Horowitz and Sartaj Sahni, 1978, Galgotia Publication
2. Introduction to Algorithms, Thomas H Cormen, Charles E Leiserson and Ronald L Rivest: 1990, TMH

Suggested Reference Books:
3. Writing Efficient Programs, Bentley, J.L., PHI
4. Introduction to Design and Analysis of Algorithm, Goodman, S.E. &Hedetnieni, 1997, MGH.

Course Outcomes:

- To identify and justify correctness of algorithms and to analyse running time of algorithms based on asymptotic analysis.
- To understand when an algorithmic design situation calls for the divide-and-conquer paradigm. Synthesize divide-and-conquer algorithms.
- Describe the greedy paradigm and dynamic-programming paradigm. Explain when an algorithmic design situation calls for it.
- Developing greedy algorithms/dynamic programming algorithms, and analyze it to determine its computational complexity.
- To write the algorithm using Backtracking and Branch and Bound strategy to solve the problems for any given model engineering problem.
PROGRAMMING IN JAVA

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PCC-CSE-309</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Course title</td>
<td>Programming in JAVA</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits  Semester 5</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Course Objectives:
- Programming in the Java programming language.
- Knowledge of object-oriented paradigm in the Java programming language.
- The use of Java in a variety of technologies and on different platforms.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit 1:
**Introduction to Java:** Evolution of Java, Object Oriented Programming Structure, Overview and characteristics of Java, Java program Compilation and Execution Process, Organization of the Java Virtual Machine, Client side Programming, Platform Independence & Portability, Security, Relation b/w JVM, JRE and JDK, Introduction to JAR format, Naming Conventions, Data types & Type casting, operators, Security Promises of the JVM, Security Architecture and Security Policy, security aspects, sandbox model

Unit 2:
**OOPS Implementation:** Classes, Objects, attributes, methods, data encapsulation, reference variables, Constructors, Anonymous block, Method Overloading, Static Data members, Block & methods; Memory Structure: Stack, Heap, Class & Method area
Class loading & Execution flow: Static vs Dynamic Class loading, implicit vs explicit class loading, class loading operations;
Argument Passing Mechanism: Passing primitive arguments, passing objects, Wrapper Classes;
This keyword: Referencing instance members, Intra class constructor chaining, Method chaining;
Inheritance & code reusability: Extending classes for code reusability, Usage of super keyword, Method Overriding, Object class

**Inheritance & Runtime Polymorphism:** Static & Dynamic binding, Inheritance and Is-A relation, Runtime Polymorphism and Generalization, Abstract classes & methods, Final Keyword;
Interfaces and Role based Inheritance: Feature & Role based Inheritance, Static & Dynamic classing Environment, classes & interfaces, interface applications in real scenarios;Has-A relation: Aggregation & Composition, Nested classes, Inner classes, Anonymous Inner classes, String Buffer Class, tokenizer, applets, Life cycle of applet and Security concerns
Unit 3:
Threads: Creating Threads, Thread Priority, Blocked States, Extending Thread Class, Runnable Interface, Starting Threads, Thread Synchronization, Synchronize Threads, Sync Code Block, Overriding Synced Methods, Thread Communication, wait, notify and notify all.

Swing & AWT:
Swing class hierarchy, containers, user interface components, graphics context, AWT Components, Component Class, Container Class, Layout Manager Interface Default Layouts, Insets and Dimensions, Border Layout, Flow Layout, Grid Layout, Card Layout Grid Bag Layout AWT Events, Event Models, Listeners, Class Listener, Adapters, Action Event Methods Focus Event Key Event, Mouse Events, Window Event

Package & Scopes: Need of Packages, associating classes to Packages, Class path environment variable, Import Keyword and Feature of static import, Public, protected, private & default scope, Private Inheritance;

Exception Handling: exception and error, Exception Handling & Robustness, Common Exceptions and Errors, Try and catch block, Exception handlers, throw keyword, Checked and Unchecked Exceptions, Role of finally, User defined Exceptions;

Unit 4:
Collection Framework: Role and Importance of Collection Framework, List & Set based collection, Iterator & List Iterator, Maps, Searching elements in List, Hash and Tree based collections, Role of equals and hashCode() methods, Comparable and Comparator Interfaces, Thread Safety and Vector, Difference b/w Enumeration and Iterator, Type safety and Generics, Common algorithms and Collections class, Using Properties class for managing properties files;

Database Connectivity Using JDBC: Overview of native and ODBC Drives, Introduction to JDBC, Type of JDBC drivers, Usage of drivers, Defining properties-based Connection Factory;

Callable Statement: Creating PL/SQL Stored procedures and functions, Creating Callable statements, Executing procedures & functions, Batch Updation, Transacting Queries, Programmatic initialization of database, ResultSetMetaData, DatabaseMetaData;

Input/Output Stream, Stream Filters, Buffered Streams, Data input and Output Stream, Print Stream Random Access File,

Reflection: reflection API, newInstance()method, javap tool, creating javap tool, creating applet viewer, call private method, java 9 features;

Text Books:
1. Patrick Naughton and HerbertzSchidt, “Java-2 the complete Reference”, TMH
2. Sierra & bates, “Head First Java”, O’Reilly.

Reference Books:
1. E. Balaguruswamy, “Programming with Java”, TMH

Course Outcomes:
- Knowledge of the structure and model of the Java programming language, (knowledge)
- Use the Java programming language for various programming technologies (understanding)
- Develop software in the Java programming language
<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-ESC-321</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Engineering Science Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Microprocessor Lab</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scheme and Credits</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
<th>Semester 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class work</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Exam</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Duration of Exam</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Hands-on experiments related to the course contents of ESC-CSE-301.
## COMPUTER NETWORKS LAB

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-CSE-323</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Computer Networks Lab</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>50 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Hands-on experiments related to the course contents of PCC-CSE-303 using hardware resources and using simulation tool.
Course Objectives:
- Implementation of various algorithms and to analyze the performance of algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

List of programs:
1. Write a Program for iterative and recursive Binary Search.
2. Write a Program to sort a given set of elements using the Quick Sort/Merge Sort/Selection Sort method and determine the time required to sort the elements.
3. Write a Program for implementation of Fractional Knapsack problem using Greedy Method and 0/1 Knapsack problem using Dynamic Programming.
4. Write a Program to find the shortest path from a given vertex to other vertices in a weighted connected graph using Dijkstra’s algorithm.
5. Write a Program to find the minimum cost spanning tree (MST) of a given undirected graph using Kruskal’s algorithm/Prim’s Algorithms.
6. Write a Program to implement N-Queens problem using back tracking.
7. Write a Program to check whether a given graph is connected or not using DFS method.
8. Write a program to implement the Travelling Salesman Problem (TSP).

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

Course Outcomes:
- The course will help in improving the programming skills of the students.
- The design of algorithms for any problem will inculcate structured thinking process in the students and improve the analytical power.
# PROGRAMMING IN JAVA LAB

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-CSE-327</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Java Programming Lab</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits  Semester 5</td>
</tr>
<tr>
<td>Classwork</td>
<td>0  0  3  1.5</td>
</tr>
<tr>
<td>Exam</td>
<td>25Marks</td>
</tr>
<tr>
<td>Total</td>
<td>50Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03Hours</td>
</tr>
</tbody>
</table>

## List of Experiments:

1. Create a java program to implement stack and queue concept.
2. Write a java package to show dynamic polymorphism and interfaces.
3. Write a java program to show multithreaded producer and consumer application.
4. Create a customized exception and also make use of all the exception keywords.
5. Convert the content of a given file into the upper case content of the same file.
6. Develop an analog clock using applet.
7. Develop a scientific calculator using swings.
8. Create an editor like MS-word using swings.
9. Create a servlet that uses cookies to store the number of times a user has visited your servlet.
10. Create a simple java bean having bound and constrained properties.
PRACTICAL TRAINING 1

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PT-CSE-329</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>PRACTICAL TRAINING 1</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Classwork: -
Exam: -
Total: -
Duration of Exam: -

The evaluation of Practical Training-I will be based on seminar, viva-voce, report submitted by the students. According to performance, the students are awarded grades A, B, C, F. A student who is awarded ‘F’ grade is required to repeat Practical Training.

Excellent: A; Good: B; Satisfactory: C; Not Satisfactory: F.
SOFTWARE ENGINEERING

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC CSE-311</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Software Engineering</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scheme and Credits</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Semester 5

Class work 25 Marks

Exam 75 Marks

Total 100 Marks

Duration of Exam 03 Hours

Objectives of the course

- Be successful professionals in the field with solid fundamental knowledge of software engineering
- Utilize and exhibit strong communication and interpersonal skills, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams
- Apply their foundations in software engineering to adapt to readily changing environments using the appropriate theory, principles and processes

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1


Software project management: Project management concepts, software process and project metrics Project planning, project size estimation metrics, project estimation Techniques, empirical estimation techniques, COCOMO- A Heuristic estimation techniques, staffing level estimation, team structures, staffing, risk analysis and management, project scheduling and tracking

Unit: 2

Requirements Analysis and specification requirements engineering, system modeling and simulation Analysis principles modeling, partitioning Software, prototyping: , Prototyping methods and tools; Specification principles, Representation, the software requirements specification and reviews Analysis Modeling: Data Modeling, Functional modeling and information flow: Data flow diagrams, BehavioralModeling; The mechanics of structured analysis: Creating entity/ relationship diagram, data flow model, control flow model, the control and process specification; The data dictionary; Other classical analysis methods.

System Design: Design concepts and principles: the design process: Design and software quality, design principles; Design concepts: Abstraction, refinement, modularity, software architecture, control hierarchy, structural partitioning, data structure, software procedure, information hiding; Effective modular design: Functional independence, Cohesion, Coupling;
Unit: 3

Architectural Design: Software architecture, Data Design: Data modeling, data structures, databases and the data warehouse, Analyzing alternative Architectural Designs, architectural complexity; Mapping requirements into a software architecture; Transform flow, Transaction flow; Transform mapping: Refining the architectural design.

Testing and maintenance: Software Testing Techniques, software testing fundamentals: objectives, principles, testability; Test case design, white box testing, basis path testing; Control structure testing: Black box testing, testing for specialized environments, architectures and applications. Software Testing Strategies: Verification and validation, Unit testing, Integration testing, Validation testing, alpha and beta testing; System testing: Recovery testing, security testing, stress testing, performance testing; The art of debugging, the debugging process debugging approaches. Software re-engineering, reverse engineering, restructuring, forward engineering.

Unit: 4


Suggested books:

- Software Engineering – A Practitioner’s Approach, Roger S. Pressman, 1996, MGH.

Suggested reference books

- Fundamentals of software Engineering, Rajib Mall, PHI
- Software Engineering Fundamentals Oxford University, Ali Behforooz and Frederick J. Hudson 1995 JW&S,
- An Integrated Approach to software engineering by Pankaj jalote , 1991 Narosa,

Course Outcomes

1. How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment
2. An ability to work in one or more significant application domains
3. Work as an individual and as part of a multidisciplinary team to develop and deliver quality software
4. Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle
5. Demonstrate an ability to use the techniques and tools necessary for engineering practice
Objectives of the course
1. Evolution of the components of system programming.
2. To learn working and different stages of compilation process.
3. To learn basic of assembler and loading schemes.
4. To learn basics of file structure.
5. To know about filters and pipeline.
6. To learn shell programming

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1
Compiler: Brief overview of compilation process, Incremental compiler, Assembler: Problem statement, symbol table; Loader schemes, compile and go Loader, general loader schemes, absolute loader, Reallocating loader, Direct linkage Loader, Binders, overlays.

Unit: 2
Theoretical Concept of Unix Operating System: Basic features of operating system; File structure: CPU scheduling; Memory management: swapping, demand paging; file system: block and fragments, inodes, directory structure; User to user communication

Unit: 3
Getting Started with Unix: User names and groups, logging in; Format of Unix commands; Changing your password; Characters with special meaning; Unix documentation; Files and directories; Current directory, looking at the directory contents, absolute and relative pathnames, some Unix directories and files; Looking at the file contents; File permissions; basic operation on files; changing permission modes; Standard files, standard output; Standard input, standard error; filters and pipelines; Processes; finding out about processes; Stopping background process; Unix editor vi.

Unit-4
Shell Programming: Programming in the Borne and C-Shell; Wild cards; Simple shell programs; Shell variables; interactive shell scripts; Advanced features.
System Administration: Definition of system administration; Booting the system; Maintaining user accounts; File systems and special files; Backups and restoration; Role and functions of a system manager. Overview of the Linux operating system

Suggested books:
1. Systems Programming by Donovan, TMH.
2. The unix programming environment by Brain Kernighen & Rob Pike, 1984, PHI & Rob Pike.
3. Design of the Unix operating system by Maurich Bach, 1986, PHI.
4. Introduction to UNIX and LINUX by John Muster, 2003, TMH.

Suggested reference books
1. Advanced Unix programmer’s Guide by Stephen Prato, BPB

Course Outcomes
1. To understand various file statistics.
2. To work on wildcards.
3. To know about shell programming and AWK utility.
Digital Image Processing

<table>
<thead>
<tr>
<th>Course Code</th>
<th>PEC-CSE-315</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Course title</td>
<td>Digital Image Processing</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L 3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course

- To become familiar with digital image fundamentals
- To get exposed to simple image enhancement techniques in Spatial and Frequency domain.
- To learn concepts of degradation function and restoration techniques.
- To study the image segmentation and representation techniques.
- To become familiar with image compression and recognition method

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1
Introduction to Image Processing: Digital Image representation, Sampling & Quantization, Steps in image Processing, Image acquisition, color image representation.

Unit: 2
Image Restoration: Image degradation and restoration process, Noise Models, Noise Filters, degradation function, Inverse Filtering, Homomorphism Filtering

Unit: 3
Image Compression: Coding redundancy, Interpixel redundancy, Psychovisual redundancy, Huffman Coding, Arithmetic coding, Lossy compression techniques, JPEG Compression.

Unit: 4
Image Segmentation & Representation: Point, Line and Edge Detection, Thresholding, Edge and Boundary linking, Hough transforms, Region Based Segmentation, Boundary representation, Boundary Descriptors.

Suggested books:
**Suggested reference books**


**Course Outcomes**

1. Know and understand the basics and fundamentals of digital image processing, such as digitization, sampling, quantization, and 2D-transforms.
2. Operate on images using the techniques of smoothing, sharpening and enhancement.
3. Understand the restoration concepts and filtering techniques.
4. Learn the basics of segmentation, features extraction, compression and recognition methods for colour models
**COMPILER DESIGN**

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PCC-CSE-302</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Compiler Design</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits Semester 6</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

**Objectives of the course:**

1. To understand and list the different stages in the process of compilation.
2. Identify different methods of lexical analysis.
3. Design top-down and bottom-up parsers.
4. Identify synthesized and inherited attributes.
5. Develop syntax directed translation schemes.

**Note:** Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

**UNIT 1**

**Introduction to Compilers:** Language Processors, The Structure of compiler: its different phases, Compiler Construction Tools, Applications of Compiler Technology.

**Lexical Analysis:** Role of lexical analyzer, Input Buffering, Specification and recognition of tokens, design of lexical analyzer, regular expressions, A language specifying lexical analyzer, Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer.

**UNIT 2**

**Syntax Analysis:** Role of parsers, context free grammars.

**Parsing Technique:** Shift-reduce parsing, Operator precedence parsing, Top down parsing, Predictive parsing.

**UNIT 3**

**LR parsers, SLR, LALR and Canonical LR parser.**

**Syntax Directed Translations:** Syntax directed definitions, construction of syntax trees, syntax directed translation scheme, implementation of syntax directed translation, Intermediate-Code Generation: three address code, quadruples and triples.
UNIT 4

Symbol Table & Error Detection and Recovery: Symbol tables: its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, Semantic error.

Code Optimization & Code Generation: Code generation, forms of objects code, machine dependent code, optimization, register allocation for temporary and user defined variables.

Suggested Text Books:


Suggested Reference Books:

2. System software by Dhamdere, 1986, MGH.
4. Elements compiler Design, Dr. M. Joseph, University Science Press

Course Outcomes:

1. To develop the lexical analyser for a given grammar specification.
2. For a given parser specification design top-down and bottom-up parsers.
3. To Develop syntax directed translation schemes.
ARTIFICIAL INTELLIGENCE

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PCC-CSE-304</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td></td>
<td>3  0  0  3</td>
</tr>
<tr>
<td>Semester</td>
<td>6</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- To provide historical perspective of AI and its foundation.
- To provide the most fundamental knowledge to the students so that they become familiar with basic principles of AI towards problem solving, inference, knowledge representation and learning.
- Explore application of AI techniques in Expert systems, Neural Networks.
- Explore the current trends, potential, limitations, and implications of AI.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction: Definition of AI, History of AI, nature of AI problems, examples of AI problems.

UNIT 2


UNIT 3


UNIT 4

Learning: Introduction to Learning, Types of Learning: Learning by Induction, Rote Learning, Symbol Based Learning, Identification Trees, Explanation Based Learning, Transformational Analogy, Introduction to Neural Networks, Expert Systems, Current trends in Artificial Intelligence
Suggested Test books:


Suggested reference books:


Course Outcomes:

1. Display the understanding of the historical perspective of AI and its foundation.
2. Apply basic principles of AI in solutions that require problem solving, inference, knowledge representation and learning.
3. Demonstrate fundamental understanding of various application of AI techniques in Expert systems, Neural Networks.
4. Demonstrate an ability to share in discussion of AI, it’s the current trends, limitations, and implications of AI.
Course code: PCC-CSE-306

Category: Professional Course Code

Course title: Advanced Java

<table>
<thead>
<tr>
<th>Scheme and Credits</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
<th>Semester 6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

Classwork: 25 Marks
Exam: 75 Marks
Total: 100 Marks
Duration of Exam: 03 Hours

**Objectives of the course:**

1. Programming in the Java programming language,
2. Knowledge of object-oriented paradigm in the Java programming language,
3. The use of Java in a variety of technologies and on different platforms.

**Note:** Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

**UNIT 1**

**Servlet:** Servlet introduction, web terminology, servlet API, servlet Interface, genericservlet, Httpservlet, servlet lifecycle, servlet with IDE (eclipse, Myeclipse, Netbeans), servlet request, servlet collaboration, servlet configuration, context, attribute in servlet, session technique in servlet, event and listener, servlet filter, CRUD, pagination, inputoutputstream, annotation, singlethread model, SSI;

**JSP:** Lifecycle of JSP, JSP API, scripting elements, 9 Implicit Objects, directive elements, Exceptions, action elements, expression language, MVC in JSP, JSTL, custom tags, pagination, CRUD, JSTL function, formatting, XML, SQL tags,

**UNIT 2**

**Struts:** Introduction, features, models, components, struts2 architecture, action, configuration, interceptors, validation method, aware Interfaces, stuts2 with I18N, zero configuration, struts2 with tiles, hibernate with struts2, spring with struts2, UI tags;

**Mail API:** Java mail introduction, methods of sending email, sending email by Gmail, receiving email, sending attachment, receiving attachment, sending HTML, forwarding, deleting email;

**UNIT 3**
**Hibernate (HB):** Introduction, architecture, HB with IDE, HB Log4j, inheritance mapping, HB mapping, transaction management, HB query language, HB criteria query language, named query, HB caching, integration, HB lifecycle;

**Spring:** Introduction, modules, spring with IDE, dependency injection methods, spring AOP, spring JdbcTemplate, spring ORM, SPEL, MVC tag library, applications, spring remoting, spring OXM, spring web, security models, spring boot, spring with angular;

**UNIT 4**

**Android:** Introduction, history & versions, architecture, building blocks, emulator, android widgets, activity and intents, android fragments, android menu, android service, SQLite, XML & JSON, android speech, multimedia, telephony, maps;

**Design Pattern:** Java design pattern, creational, structural, behavioral, J2EE patterns, presentation layers,

**Course Outcome:**

1. Knowledge of the structure and model of the Java programming language, (knowledge)
2. Use the Java programming language for various programming technologies (understanding)
3. Develop software in the Java programming language,

**Suggested Text Books:**

1. Patrick Naughton and Herbert Schidt, “Java-2 the complete Reference”, TMH

**Suggested Reference Books:**

1. E. Balaguruswamy, “Programming with Java”, TMH
DATA SCIENCE

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>ESC-CSE-308</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Engineering Science Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Data Science</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td></td>
</tr>
<tr>
<td></td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- The objective of this course is to impart necessary knowledge of the basic foundations needed for understanding data science domain and develop programming skills required to build data science applications.
- To introduce the conceptual knowledge of the area of data science domain, feature and scope of applications.
- To impart programming knowledge needed for data sciences.
- To understand the different issues involved in the design and implementation of a data science applications.
- To understand case studies of essential Data sciences applications.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction to Data Science: Concept of Data Science, Traits of Big data, Web Scraping, Analysis vs Reporting, Collection, storing, processing, describing and modelling, statistical modelling and algorithm modelling, AI and data science, Myths of Data science

UNIT 2

Introduction to Programming Tools for Data Science: Toolkits using Python: Matplotlib, NumPy, Scikit-learn, NLTK, Visualizing Data: Bar Charts, Line Charts, Scatterplots, Working with data: Reading Files, Scraping the Web,

UNIT 3

Data Science Methodology: Business Understanding, Analytic Approach, Data Requirements, Data Collection, Data Understanding, data Preparation, Modeling, Evaluation, Deployment, feedback
UNIT 4

Data Science Application: Prediction and elections, Recommendations and business analytics, clustering and text analytics

Suggested Text books:
1. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media

Suggested Reference books:
1. Data Science Workflow: Overview and Challenges by Philip Guo
2. Python for Data Analysis, O’Reilly Media.
5. Jiawei Han and Jian Pei, "Data Mining Concepts and Techniques", Third Edition, Morgan Kaufmann Publishers

Course Outcomes:
- Understand the value of data science and the process behind using it.
- Use Python to gather, store, clean, analyse, and visualise data-sets.
- Apply toolkits to formulate and test data hypotheses and uncover relationships within data-sets
- Understand the data science methodology in the data science pipeline
- Understand real-world challenges with several case studies
Students will be assigned projects individually or in a group of not more than 3 students depending on the efforts required for completion of project.

The project will have 4 stages:
(*Marks for internal evaluation are given in brackets)

- Synopsis submission (5 marks),
- 1st mid term progress evaluation (5 marks)
- 2nd mid term progress evaluation (5 marks)
- Final submission evaluation (10 marks).

The external examiner will evaluate the project on the basis of idea/quality of project, implementation of the project, project report and viva.
COMPILERDESIGN LAB

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-CSE-324</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>CompilerDesign Lab</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>50 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:
- Implementation of different concepts of lexical analysis.
- Implementation of parsers.
- Study and use of compiler design tools.

List of programs:
1. Write a Program for Token separation with a given expression.
2. Write a Program for Token separation with a given file.
3. Write a Program for Lexical analysis using LEX tools.
4. Write a Program to identify whether a given line is a comment or not.
5. Write a Program to check whether a given identifier is valid or not.
6. Write a Program to recognize strings under 'a', 'a*b+', 'abb'.
7. Write a Program to simulate lexical analyser for validating operators.
8. Write a Program for implementation of Operator Precedence Parser.
9. Study of LEX and YACC tools:
   i) Write a Program for implementation of calculator using YACC tool.
   ii) Write a Program for implementation of Recursive Descent Parser using LEX tool.
10. Write a Program for implementation of LL (1) Parser.
11. Write a Program for implementation of LALR Parser

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

Course Outcomes:
- The course will help in improving the programming skills of the students.
- The implementation of different parsers will help in understanding of compiler designing.
ARTIFICIAL INTELLIGENCE LAB USING PYTHON

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-CSE-326</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Course title</td>
<td>Artificial Intelligence Lab Using Python</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scheme and Credits</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1.5</td>
</tr>
<tr>
<td>Semester 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Class work 25 Marks
Exam 25 Marks
Total 50 Marks
Duration of Exam 03 Hours

List of Program:

1. Write a Program to Implement Breadth First Search using Python.
2. Write a Program to Implement Depth First Search using Python.
3. Write a Program to Implement Tic-Tac-Toe game using Python.
4. Write a Program to Implement 8-Puzzle problem using Python.
5. Write a Program to Implement Water-Jug problem using Python.
6. Write a Program to Implement Travelling Salesman Problem using Python.
7. Write a Program to Implement Tower of Hanoi using Python.
8. Write a Program to Implement Monkey Banana Problem using Python.
10. Write a Program to Implement 8-Queens Problem using Python.

Note: At least 5 to 10 more exercises to be given by the teacher concerned.
## ADVANCED JAVA LAB

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>LC-CSE-328</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Advanced Java Lab</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td></td>
<td>0   0  2   1</td>
</tr>
<tr>
<td>Semester</td>
<td>6</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>50 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Students have to write at least 15 programs based on the course PCC-CSE-306
Course Objectives:

Students will be able to:

1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals’ constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

Unit – I

Philosophy of Indian Constitution: Salient features of Indian Constitution, Preamble, and Nature of Indian Constitution, Procedure for amendment of the Constitution.

Unit – II

Federal structure and distribution of legislative and financial powers between the Union and the States

Unit – III


Unit – IV

Fundamental Rights: Origin and development of Fundamental rights, Need for fundamental rights. Introduction to Right to equality, Right to freedom, Right against exploitation, Right to freedom of religion, Cultural and Education rights and Fundamental duties.

References:

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S.N. Busi, Dr. B.R. Ambedkar framing of Indian Constitution, latest Edition

Course Outcomes:

Students will be able to:
1. Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.

2. Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.

3. Discuss the circumstances surrounding the foundation of the Congress Socialist Party (CSP) under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.


The examination of the regular students will be conducted by the concerned college/Institute internally. Each student will be required to score minimum 40% marks to qualify in the paper. The marks will not be included in determining the percentage of marks obtained for the award of degree. However, these marks will be shown in the detailed marks certificate of the students.
ADVANCED DATABASE MANAGEMENT SYSTEM

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC-CSE-310</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Advanced Database Management System</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objective of the course:

- To understand DBMS Components, Advantages and Disadvantages.
- Understanding Data modeling: ER, EER, Network, Hierarchical and Relational data models.
- Understanding normalization, general strategies for query processing, query processor, syntax analyzer, Query decomposition, Heuristic Query optimization.
- To understand transaction concept, schedules, serializability, locking and concurrency control protocols.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction: Architecture, Advantages, Disadvantages, Data models, relational algebra, SQL, Normal forms.

Query Processing: General strategies for query processing, transformations, expected size, statistics in estimation, query improvement. Query evaluation, view processing, query processor.

UNIT 2

Recovery: Reliability, Transactions, recovery in centralized DBMS, reflecting updates, Buffer management logging schemes, disaster recovery.

Concurrency: Introduction, Serializability, Concurrency control, Locking schemes, Timestamp based ordering, Optimistic, Scheduling, Multiversion techniques, Deadlocks.

UNIT 3

Parallel and Distributed Databases: Distributed Data Storage – Fragmentation & Replication, Location and Fragment. Transparency Distributed Query Processing and Optimization, Distributed Transaction Modeling and concurrency Control, Distributed Deadlock, Commit Protocols, Design of Parallel Databases, Parallel Query Evaluation.

UNIT 4
Objected Oriented and Object Relational Databases: Modeling Complex Data Semantics, Specialization, Generalization, Aggregation and Association, Objects, Object Identity, Equality and Object Reference, Architecture of Object Oriented and Object Relational Databases

Suggested Text Book:


Suggested References Books:


Course Outcomes:

- Students will get understanding of DBMS Components, its advantages and disadvantages.
- Understanding about various types of Data modeling: ER, EER, Network, Hierarchical and Relational data models.
- Understanding normalization, general strategies for query processing, query processor, syntax analyzer, Query decomposition, Heuristic Query optimization.
- Understanding transaction concept, schedules, serializability, locking and concurrency control protocols.
MOBILE APPLICATIONS DEVELOPMENT

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC-CSE-312</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Mobile applications development</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td>Class work</td>
<td>3</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- Introduce the students with the various “Next Generation Technologies” in the area of mobile computing
- Assist students understand the various Mobile operating Systems
- Explore the findings using Android Technologies

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction: Mobile operating system, Operating system structure, Constraints and Restrictions, Hardware configuration with mobile operating system, Features: Multitasking Scheduling, Memory Allocation, File System Interface, Keypad Interface, I/O Interface, Protection and Security, Multimedia features

UNIT 2

Introduction to Mobile development IDE's, Introduction to Worklight basics, Optimization, pages and fragments , Writing a basic program- in Worklight Studio, Client technologies, Client side debugging, Creating adapters, Invoking adapters from Worklight Client application, Common Controls, Using Java in adapters, Programming exercise with Skins, Understanding Apache Cordova.

UNIT 3


UNIT 4

Android: Introduction to Android, Architecture, memory management, communication protocols, application development methods, deployment.
iOS: Introduction to iOS, Architecture, memory management, communication protocols, application development methods, deployment

Suggested text books:

1. Anubhav Pradhan, Anil V Deshpande, “Mobile Apps Development” Edition:
4. Teach Yourself Android Application Development In 24 Hours, Edition: I, Publication: SAMS

Suggested reference books:

1. Neal Goldstein, Tony Bove, “iPhone Application Development All-In-One For Dummies”, John Wiley & Sons

Course Outcomes:

• Explain the principles and theories of mobile computing technologies.
• Describe infrastructures and technologies of mobile computing technologies.
• List applications in different domains that mobile computing offers to the public, employees, and businesses.
• Describe the possible future of mobile computing technologies and applications.
• Effectively communicate course work through written and oral presentations.
COMPUTER GRAPHICS

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC-CSE-314</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Computer Graphics</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td></td>
<td>3   0   0   3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- To have basic understanding of the core concepts of Computer Graphics.
- Understand scan conversion, 2D, 3D – transformation and viewing.
- To be able to create interactive computer Graphics with understanding of shading, image processing and illumination model.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1


UNIT 2

Two dimensional transformations: Geometric, Coordinate and, composite transformation. Two Dimensional Viewing: window to view port mapping; Clipping: point, line, polygon, curve and text clipping

UNIT 3


UNIT 4

**Illumination, shading, image manipulation:** Illumination models, shading models for polygons, shadows, transparency, image processing.

**Suggested Text Books:**


**Suggested Reference Books:**


**Course Outcomes:**

- Understanding of the software, hardware and applications of Computer Graphics.
- Understanding of Scan conversion, 2D, 3D – transformation and viewing.
- To be able to implement picture on screen using projection, shading, image processing and illumination model.
COMMUNICATION ENGINEERING

<table>
<thead>
<tr>
<th>Course code</th>
<th>PEC-CSE-330 (Common with ECE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Program Elective Course</td>
</tr>
<tr>
<td>Course title</td>
<td>Communication Engineering</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td></td>
<td>3   0   0   3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Course Objective:

1. The course will give students about depth knowledge of the communication system.
2. To introduce students to random process and fundamental theorems.
3. To make awareness of information theory and coding techniques.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit:1
SPECTRAL ANALYSIS:
Fourier series, Fourier transforms, Convolution Theorem, Correlation, Cross-Correlation and autocorrelation.

Unit: 2
INFORMATION THEORY:
Introduction to information and entropy, channel capacity for discrete and continuous channels, Shannon’s Theorem, Shannon-Hartley Theorem, Noisy channels, coding theory: Shannon-Fano coding, minimum redundancy coding, maximization of entropy of a continuous message transmission rate, effect of medium on the information, selection of channels, effect of noise and its minimization.

Unit:3
RANDOM SIGNAL THEORY:
Representation of random signals, concept of probability, probability of joint occurrence, conditional probability, discrete probability theory, continuous random variables, probability distribution function, probability density function, joint probability density functions.

Unit:4
RANDOM SIGNAL THEORY:
Statistical average and moments, Ergodic processes, correlation Function, power spectral density, central limit theory, response of linear system to random signals. Error function Covariance relation among the spectral densities of the two input-output random processes. Cross spectral densities, optimum filters. Introduction to Linear Block Code and cyclic Codes.
TEXT BOOK:
1. Principles of Communication Systems: Taub Schiling; TMH

REFERENCE BOOKS.
1. Communication Systems: Singh and Sapre ; TMH
2. Communication Systems: A Bruce Carlson; TMH

COURSE OUTCOMES: After the completion of the course the student will be able to:

- To Study and Derive equations for entropy mutual information and channel capacity for all types of channels.
- To acquire the knowledge about Fourier series and Fourier transform signal analysis tool.
- Design a digital communication system by selecting an appropriate error correcting codes for a particular application.
- To learn about Probability of Random signal theory and process.
- Formulate the basic equations of linear block codes and a cyclic code.
- Compare the performance of digital communication system by evaluating the probability of error for different error correcting codes.
DISTRIBUTED SYSTEM

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC-CSE-316</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Elective Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Distributed System</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L  T  P  Credits</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- To examine the fundamental principles of distributed systems, and provide students hands-on experience in developing distributed protocols.
- Analyze the issues in distributed operating systems and to address these distributed systems issues in a broader sense. Emphasis will be placed on communication, process, naming, synchronization and fault tolerance.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1

Introduction: Distributed Operating Systems Definition and goals, Hardware and Software concepts, Design issues.

Communication in Distributed System: Computer Network and Layered protocols, Message passing and related issues, synchronization, Client Server model & its implementation, remote procedure call and implementation issues, Case Studies: SUN RPC, DEC RPC

UNIT 2

Synchronization in Distributed System: Clock synchronization and related algorithms, mutual exclusion, Deadlock in distributed systems

Processes and processors in Distributed systems: Threads, system model, processor allocation, scheduling in distributed systems: Load balancing and sharing approach, fault tolerance, real time distributed systems, Process migration and related issues

UNIT 3

Distributed File systems: Introduction, features & goal of distributed file system, file models, file accessing models, file sharing semantics, file caching scheme, file replication, fault tolerance, trends in distributed file system, case study.
Distributed Shared Memory: Introduction, general architecture of DSM systems, design and implementation issues of DSM, granularity, structure of shared memory space, consistency models, replacement strategy, thrashing

UNIT 4


Distributed Web-based Systems: Architecture, Processes, Communication, Naming, Synchronization

Case Studies: JAVA RMI, Sun Network File System, Google Case Study

Suggested Reference books:

1. Distributed Operating Systems by Andrew S Tannebaum, Pearson
4. Distributed Computing by Sunita Mahajan & Seema Shah OXFORD
5. Distributed Systems: Principles and Paradigms by Andrew S Tanebaum, Maarten Van Steen, PHI
6. Distributed Computing, Fundamentals, Simulations and Advanced topics, 2nd Edition, Hagit Attiya and Jennifer Welch, Wiley India

Course Outcomes:

• List the principles of distributed systems and describe the problems and challenges associated with these principles.
• Understand Distributed Computing techniques, Synchronous and Processes.
• Apply Shared Data access and Files concepts.
• Design distributed system that fulfills requirements with regards to key distributed systems properties.
• Understand Distributed File Systems and Distributed Shared Memory.
• Apply Distributed web-based system and understand the importance of security in distributed system
INFORMATION TECHNOLOGY & INDUSTRY BUSINESS SKILLS

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PEC-CSE-318</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Information Technology &amp; Industry Business Skills</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td>Semester 6</td>
<td>3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:
- To understand the novel information technology techniques and industry business skills.
- To study about the concept of amazon web services.
- To understand the use of cloud in web services and their different application.
- To study business models used in industry and their implementation.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

Unit: 1

Unit: 2
Cloud: Amazon Elastic Compute Cloud, Elastic Block Store, Security Group management, Amazon Machine Images, Storing data in the cloud, storing your objects: S3 and Glacier, ELB and SQS, auto-scaling and Cloud Watch, Cloud Formation, Elastic Beanstalk, and Ops Works, RDS, fault-tolerance, scaling, AZURE architecture and services, Google cloud applications;

Unit: 3
Business: Business models, Building blocks of Sales force, Understand the Security model, Understand the Data model, Configure and manage Sales and Service Cloud, Learn about Sales force Objects, create, rename or modify objects, validation rules, Create different field types and validation rules, Sales Cloud and Service Cloud modules, reports and dashboard, Sales force A Chatter, and Social features, chatter, application lifecycle, visual workflow;
Unit: 4

Security & Applications: security group, NACL, difference between security group and NACL, AWS-Data pipeline, Simple queue services, Simple workflow services, Simple notification Services, Elastic Transcoder.

Suggested reference books:

1. Amazon Web Services in Action by Michael Wittig and Andreas Wittig, Manning Publications;
2. AWS Certified Solutions Architect by Joe Baron, Hisham Baz, Tim Bixler, Biff Gaut, Kevin E. Kelly, Wiley publication;

Course Outcomes:

- Student will understand the concept of web services of amazon, virtual machines and their working.
- For a given region the availability of resources and cost management.
- For a given application scalable model and selection of services.
MOBILE AND WIRELESS COMMUNICATION

<table>
<thead>
<tr>
<th>Coursecode</th>
<th>PCC-CSE-320</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Professional Core Course</td>
</tr>
<tr>
<td>Coursetitle</td>
<td>Mobile and wireless communication</td>
</tr>
<tr>
<td>Scheme and Credits</td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Class work</td>
<td>25 Marks</td>
</tr>
<tr>
<td>Exam</td>
<td>75 Marks</td>
</tr>
<tr>
<td>Total</td>
<td>100 Marks</td>
</tr>
<tr>
<td>Duration of Exam</td>
<td>03 Hours</td>
</tr>
</tbody>
</table>

Objectives of the course:

- Understand the wireless/cellular radio concepts such as frequency reuse, handoff and interference between mobiles and base stations.
- Identify the techno-political aspects of wireless and mobile communications such as the allocation of the limited wireless spectrum by regulatory agencies.
- Understand the information theoretical aspects such as channel capacity, propagation effects, modeling the impact of signal bandwidth and motion in mobile systems.
- Describe the mobility support mechanism, WWW and WAPs.

Note: Examiner will set nine questions in total. Question one will be compulsory. Question one will have 6 parts of 2.5 marks each from all units and remaining eight questions of 15 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, first being compulsory and selecting one from each unit.

UNIT 1


MAC Layer: Specialized MAC, SDMA, FDMA, TDMA – Fixed TDM, Classical ALOHA, Slotted ALOHA, CSMA, DAMA, PKMA, Reservation TDMA. Collision Avoidance, Polling, Inhibit Sense Multiple Access, CDMA.


UNIT 2

**Wireless LAN:** IEEE 802 11- System and Protocol Architecture, Physical Layer, MAC Layered Management.

**Bluetooth:** User scenarios, Physical layer, MAC Layer, Networking, Security and Link Management.

**Wimax**

**UNIT 3**

**Mobile Network Layer:** Mobile IP-Goals, Assumptions, Requirement, Entities, Terminology, IP Packet delivery, Agent Advertisement and Discovery, Registration, Tunneling, Encapsulation, Optimization, Reserve Tunneling, Security, IPv6, DHCP.

**Mobile Adhoc Networks:** Routing, Destination Sequence Distance Vector, Dynamic Source Routing, Hierarchical algorithms, Performance Metrics.

**Mobile Transport Layer:** Traditional TCP, Indirect TCP, Snooping, TCP, Mobile TCP, Fast-retransmission TCP, Transaction oriented TCP.

**UNIT 4**

**Satellite Systems:** History, Applications, GEO, LEO, MEO, Routing, Localization, Handover in Satellite System.

**Support for Mobility:** File System, WWW, HTML, System Architecture.


**Suggested Reference Books:**


**Course Outcomes:**

- Explain the principles and theories of mobile computing technologies.
- Describe infrastructures and technologies of mobile computing technologies.
- List applications in different domains that mobile computing offers to the public, employees, and businesses.
- Describe the possible future of mobile computing technologies and applications.
- Effectively communicate course work through written and oral presentations.