

Computer Science & Engineering and Information Technology					
Code: CS501	Operating System	L	T	P	C
		3	1	0	4

**Course Outcome:**

1. Apply optimization techniques for the improvement of system performance.
2. Ability to design and solve synchronization problems.
3. Learn about minimization of turnaround time, waiting time and response time and also maximization of throughput by keeping CPU as busy as possible.
4. Ability to change access controls to protect files.
5. Ability to compare the different operating systems.

**CO-PO Mapping:**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
<b>CS5103.1</b>	2	2	2	2	-	-	-	-	1	-	3	1
<b>CS5103.2</b>	2	-	3	-	-	-	-	-	2	-	1	2
<b>CS5103.3</b>	1	2	3	2	2	-	-	-	3	-	1	2
<b>CS5103.4</b>	-	-	-	-	-	-	-	-	1	-	1	-
<b>CS5103.5</b>	-	1	-	-	2	-	-	-	-	-	1	-
<b>Average</b>	1	1	1.4	0.8	0.8	0	0	0	1.4	0	1.4	1

**MODULE 1: FUNDAMENTALS**

**Overview:** Role and purpose of operating systems; history of operating system development; functionality of a typical operating system; design issues (efficiency, robustness, flexibility, portability, security, compatibility).

**Basic principles:** Structuring methods; abstractions, processes, and resources; design of application programming interfaces (APIs); device organization; interrupts; user/system state transitions.

**MODULE 2: PROCESS MANAGEMENT**

**Scheduling:** Preemptive and non-preemptive scheduling; scheduling policies; processes and threads; real-time issues;

**Concurrency:** The idea of concurrent execution; states and state diagrams; implementation structures (ready lists, process control blocks, and so forth); dispatching and context switching; interrupt handling in a concurrent environment;

**Mutual exclusion:** Definition of the “mutual exclusion” problem; deadlock detection and prevention; solution strategies; models and mechanisms (semaphores, monitors, condition variables, rendezvous); producer-consumer problems; synchronization; multiprocessor issues

**MODULE 3: MEMORY MANAGEMENT**

Review of physical memory and memory management hardware; overlays, swapping, and partitions; paging and segmentation; page placement and replacement policies; working sets and thrashing;

caching.

#### **MODULE 4: SECONDARY STORAGE MANAGEMENT**

**Device management:** Characteristics of serial and parallel devices; abstracting device differences; buffering strategies; direct memory access; recovery from failures.

**File systems:** Fundamental concepts (data, metadata, operations, organization, buffering, sequential vs non-sequential files); content and structure of directories; file system techniques (partitioning, mounting and un-mounting, virtual file systems); memory-mapped files; special-purpose file systems; naming, searching, and access; backup strategies.

#### **MODULE 5 SECURITY AND PROTECTION**

Overview of system security; policy/mechanism separation; security methods and devices; protection, access, and authentication; models of protection; memory protection; encryption; recovery management.

#### **Text / Reference Books**

1. Silberschatz, P.B. Galvin & G. Gagne, Operating .system concepts, John Wiley,9th Edition,2012
2. W. Stallings, Operating systems, Prentice-Hall, 2012

Computer Science & Engineering and Information Technology							
<b>Code: CS502</b>	<b>Database Management System</b>			<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
				<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

### Course Outcomes:

1. Realize the difference between file system and database system and appreciate the features of the database management system.
2. Must be able to model application requirements using the tools like ER modelling and database schemas.
3. Should be familiar with the basic operations of SQL and their constraints and their utility at the application level.
4. Must be aware of the normalization techniques and their applications for database design.
5. Should have an idea about the issues involved during processing of transactions, their solutions and data preservation.

### CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	-	-	-	2	-	-	-	-	-	-	2
CO.2	-	2	2	1	-	2	2	-	-	-	1	-
CO.3	-	1	-	-	3	-	-	-	-	-	1	-
CO.4	-	1	2	1	-	-	-	-	-	-	-	-
CO.5	2	2	3	-	-	-	2	-	-	-	-	-

### MODULE 1: INTRODUCTION

File & Data Base Concept, Overview of DBMS, Data Models, Schema and Instances, Data Independence

### MODULE 2: ENTITY-RELATIONSHIP MODEL

Basic concepts, Keys, Entity-Relationship Diagram, Cardinality ratios, Strong & Weak Entity Sets, Specialization, Generalization, Aggregation.

Relational Model

Procedural & Non Procedural Languages, Relational Algebra, Extended Relational Algebra Operations, Views, Modifications Of the Database, Relational Calculus.

### MODULE 3: SQL

Basic Concepts, Set operations, Aggregate Functions, Null Values, assertions, views, Nested Sub-queries, Cursors, Stored procedures and triggers.

Integrity Constraints & Introduction to RDBMS

Domain Constraints, Referential Integrity Constraints, Codd's rule.

#### **MODULE 4: FUNCTIONAL DEPENDENCIES AND NORMALIZATION**

Functional Dependency, Armstrong's axioms, Canonical Cover, Closure, Full and Partial Functional dependencies, Prime & Non Prime attribute, 1NF, 2NF, 3NF, BCNF, Multi valued Dependency , 4NF, 5NF, DKNF.

#### **MODULE 5: TRANSACTION & CONCURRENCY CONTROL**

Transaction concept, ACID properties, Conflict & View serializability, Test for Conflict serializability, Concurrency Control, Lock base protocols, Two phase locking.

#### **MODULE 6: STORAGE STRATEGIES**

Single-Level Index (primary, secondary, clustering), Multi-level Indexes, Dynamic Multi-level Indexes, Hashing Techniques, B tree and B+ tree.

Query Optimization

Full Table scan, Indexed-based scan, Merge join, Nested loop join, Equivalence rules , Heuristic Optimization , Cost Based Optimization.

#### **MODULE 7: BACKUP & RECOVERY**

Physical & Logical Backup, Transaction logs, Causes of failures, Recovery techniques.

Distributed Databases

Basic Concepts, Data Fragmentation, Replication and Allocation Techniques, Types of Distributed Database Systems, Query Processing, Overview of Client-Server Architecture and Its relationship to Distributed Databases.

#### **TEXT BOOKS:**

1. Data base System Concepts, Silberschatz, Korth, McGraw Hill
2. Fundamentals of Database Systems, Elmasri Navathe Pearson Education

#### **REFERENCES:**

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Introduction to Database Systems, C.J. Date Pearson Education

Computer Science & Engineering and Information Technology					
Code: CSE511	Compiler Design	L	T	P	C
		3	0	0	3

## COURSE OUTCOMES

1. *Identify* the issue that arises in the design and construction of translator for programming language.
2. *Analyze* RE and CFG to specify the lexical and syntactic structure of programming language.
3. *Design* different parsers from given specification.
4. *Assess* the various program transformations.
5. *Design* a compiler for a programming language.

### MODULE 1: INTRODUCTION TO COMPILER

Compilers, Analysis of source programs, Tokens, patterns, lexemes, Phases of compilers, Parsing, Parse trees, Ambiguity, Associativity and precedence of operators, Top-down parsing, Bottom-up parsing, Left recursion, Syntax directed translation.

### MODULE 2: FINITE AUTOMATA

Classification of grammars, NFA, DFA, Conversion of NFA to DFA, RE to NFA (Thompson's Construction), Optimization of NFA/DFA using FIRSTPOS, LASTPOS, FOLLOWPOS.

### MODULE 3: CONTEXT FREE GRAMMAR

RE vs. CFG, Eliminating ambiguity and left recursion, Left factoring.

### MODULE 4: COMPILER PARSER

Top down parsing-LL parser, LL grammars. Bottom up parsing- LR parser, SLR parser, CLR parser, LALR parser. Polishing expressions Operator precedence grammar. LR grammars. Comparison of parsing methods. Error handling.

### MODULE 5: RUN TIME ENVIRONMENTS

Symbol tables, Language facilities for dynamic storage allocation, Dynamic storage allocation technique, Organization for non-block and block structured languages.

### MODULE 6: INTERMEDIATE CODE GENERATION

Intermediate languages, graphical representations, Synthesized and inherited attributes, Dependency graph, Syntax directed translation, S and L- attributed definitions, Polish notation, Three address, quadruples, triples, indirect triples Flow of control statement.

## **MODULE 7: CODE OPTIMIZATION AND CODE GENERATION**

Basic blocks and flow graphs, Optimization of basic blocks, Code optimization techniques, Issues in design of code generator, Target machine code and simple code generator.

### **TEXT BOOKS**

- Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Monica S. Lam, *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 2006 (optional).
- Thomas W. Parsons, *Introduction to Compiler Construction*. Computer Science Press, **1992**.

### **REFERENCE BOOKS**

- Compiler design in C, A.C. Holub, PHI.
- Compiler construction (Theory and Practice), A.Barret William and R.M. Bates, Galgotia Publication.
- Compiler Design, Kakde.

Computer Science & Engineering and Information Technology					
<b>Code: IT511</b>	<b>Internetworking</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

### Course Outcomes:

CO 1: Students will be able to classify the routing protocols and analyse how to assign the IP addresses for the given network.

CO 2: Students will be able to understand the architecture of different internet servers.

CO 3: Students will be able to configure the firewall in the network.

### MODULE 1:- AN OVERVIEW ON INTERNET

The need for an Internet, The TCP/IP Internet, Internet services, Internet protocols and standardization, Review of Network technologies.

### MODULE 2: - INTERNETWORKING CONCEPTS

Architectural model introduction, Application level interconnection, Network level interconnection, Properties of the Internet, Internet Architecture, Interconnection through IP Gateways or routers, Internet and Intranet.

### MODULE 3: - INTERNET ADDRESS

Introduction, Universal identifiers, Three primary classes of IP addresses, Classless IP address, Network and Broadcast addresses, Mapping internet addresses to physical addresses (ARP), ARP protocol format, Transport Gateways and subnet addressing, Multicast addressing.

### MODULE 4: - INTERNET PROTOCOL

Internet Architecture and Philosophy, The concept of unreliable delivery, Connectionless delivery system, The Internet Datagram, Routing direct and indirect delivery, Table driven IP routing, Protocol layering, Reliable stream transport, TCP performance, Bootstrap protocol (BOOTP).

### MODULE 5: - ROUTING

The origin of Gateway routing tables, Original Internet Architecture and Cores, Core Gateways, Automatic route propagation, Vector distance (Bellman-Ford), routing, Gateway to Gateway Protocol (GGP), Autonomous system concept, Exterior Gateway Protocol (EGP), Interior Gateway Protocol (RIP, OSPF, HELLO), Routing Information Protocol (RIP), Combining RIP, HELLO, and EGP, Routing with partial information.

### MODULE 6: - ENTERPRISE NETWORKING AND INTERNET SERVERS

Corporate networking, Broadband at the Metropolitan area level, High speed dedicated WAN services and switched WAN services, ISDN, BISDN and ATM services, Frame relay technology and services, Virtual private network concepts PPTP protocol. DNS, DHCP Servers, FTP, TELNET, E-Mail.

### MODULE 7: - FIREWALL & NETWORKING

Introduction, Implementation of Firewall, Activities of Firewall, Configuration of firewall, Firewalls & SSL, SSL implementation, Bit implementation of SSL, Use of SSL.

### **REFERENCE BOOKS**

1. Computer Networks and Internets - Douglas E. Comer; PE.
2. Communication Networks - Leon-Garcia-Widjaja; TMH.
3. Internetworking with TCP / IP - Douglas E .Comer; PE.
4. TCP/IP protocol suite - Forouzan Behrouz A; TMH.
5. Computer Networks – Andrew S. Tanenbaum; PHI.
6. Data and Computer Communication - William Stallings; PHI.
7. The Complete reference of Networking - Craig Zacker; TMH.



Computer Science & Engineering and Information Technology					
Code: CS512	System Analysis and Design	L	T	P	C
		3	0	0	3

## COURSE OUTCOME

1. To provide a solid foundation of systems principles
2. An understanding of how business function, while heightening students to the issues analysts face daily.

## COURSE CONTENTS:

### MODULE 1: INTRODUCTION

**System definition and concepts:** Characteristics and types of system, Manual and automated systems

**Real-life Business sub-systems:** Production, Marketing, Personal, Material, Finance

**Systems models types of models:** Systems environment and boundaries, Real-time and distributed systems, Basic principles of successful systems

### MODULE 2: SYSTEMS ANALYST

Role and need of systems analyst, Qualifications and responsibilities, Systems Analyst as and agent of change,

**Introduction to systems development life cycle (SDLC):**

**Various phases of development:** Analysis, Design, Development, Implementation, Maintenance

**Systems documentation considerations:** Principles of systems documentation, Types of documentation and their importance, enforcing documentation discipline in an organization.

**System Planning**

Data and fact gathering techniques: Interviews, Group communication, Presentations, Site visits.

Feasibility study and its importance, Types of feasibility reports System Selection plan and proposal Prototyping

**Cost-Benefit and analysis:** Tools and techniques

### MODULE 3: SYSTEMS DESIGN AND MODELING

Process modeling, Logical and physical design, Design representation, Systems flowcharts and structured charts, Data flow diagrams, Common diagramming conventions and guidelines using DFD and ERD diagrams. Data Modeling and systems analysis, designing the internals: Program and Process design, Designing Distributed Systems.

**Input and Output Classification of forms:** Input/output forms design, User-interface design, Graphical interfaces

### MODULE 4: MODULAR AND STRUCTURED DESIGN

Module specifications, Module coupling and cohesion, Top-down and bottom-up design

**.System Implementation and Maintenance**

Planning considerations, Conversion methods, producers and controls, System acceptance Criteria, System evaluation and performance, Testing and validation, Systems qualify Control and assurance, Maintenance activities and issues.

### MODULE 5: SYSTEM AUDIT AND SECURITY

**Computer system as an expensive resource:** Data and Strong media Procedures and norms for utilization of computer equipment, Audit of computer system usage, Audit trails,

**Types of threats to computer system and control measures:** Threat to computer system and control measures, Disaster recovery and contingency planning

**Object Oriented Analysis and design**

Introduction to Object Oriented Analysis and design life cycle, object modeling: Class Diagrams,

Dynamic modeling: state diagram, Dynamic modeling: sequence diagramming.

**TEXT BOOKS/ REFERENCES: -**

1. System Analysis and Design Methods, Whitten, Bentley and Barlow, Galgotia Publication.
2. System Analysis and Design Elias M. Award, Galgotia Publication
3. Modern System Analysis and Design, Jeffrey A. Hofer Joey F. George Joseph S. Valacich Addison Weseley.

Computer Science & Engineering and Information Technology					
Code: EC524	Analog and Digital Communication	L	T	P	C
		3	0	0	3

## COURSE OUTCOME:

After completing this course, the students will be able to:

- 1) Understand analog and digital communication techniques.
- 2) Learn data and pulse communication techniques.
- 3) Be familiarized with source and Error control coding.
- 4) Gain knowledge on multi-user radio communication.

## SYLLABUS:

**Module 1: ANALOG COMMUNICATION:** Noise: Source of Noise – External Noise- Internal Noise- Noise Calculation. Introduction to Communication Systems: Modulation – Types – Need for Modulation. Theory of Amplitude Modulation – Evolution and Description of SSB Techniques – Theory of Frequency and Phase Modulation – Comparison of various Analog Communication System (AM – FM – PM).

**Module 2: DIGITAL COMMUNICATION :** Amplitude Shift Keying (ASK) – Frequency Shift Keying (FSK) Minimum Shift Keying (MSK) –Phase Shift Keying (PSK) – BPSK – QPSK – 8 PSK – 16 PSK – Quadrature Amplitude Modulation (QAM) – 8 QAM – 16 QAM – Bandwidth Efficiency– Comparison of various Digital Communication System (ASK – FSK – PSK – QAM).

**Module 3: DATA AND PULSE COMMUNICATION :** Data Communication: History of Data Communication – Standards Organizations for Data Communication- Data Communication Circuits – Data Communication Codes – Error Detection and Correction Techniques – Data communication Hardware – serial and parallel interfaces. Pulse Communication: Pulse Amplitude Modulation (PAM) – Pulse Time Modulation (PTM) – Pulse code Modulation (PCM) – Comparison of various Pulse Communication System (PAM – PTM – PCM).

**Module 4: SOURCE AND ERROR CONTROL CODING:** Entropy, Source encoding theorem, Shannon fano coding, Huffman coding, mutual information, channel capacity, channel coding theorem, Error Control Coding, linear block codes, cyclic codes, convolution codes, viterbi decoding algorithm.

**Module 5: MULTI-USER RADIO COMMUNICATION:** Advanced Mobile Phone System (AMPS) – Global System for Mobile Communications (GSM) – Code division multiple access (CDMA) – Cellular Concept and Frequency Reuse – Channel Assignment and Hand – Overview of Multiple Access Schemes – Satellite Communication – Bluetooth.

**TEXT BOOK:**

1. Wayne Tomasi, "Advanced Electronic Communication Systems", 6th Edition, Pearson Education, 2009.

**REFERENCES:**

1. Simon Haykin, "Communication Systems", 4th Edition, John Wiley & Sons, 2004
2. Rappaport T.S, "Wireless Communications: Principles and Practice", 2nd Edition, Pearson Education, 2007
3. H.Taub, D L Schilling and G Saha, "Principles of Communication", 3rd Edition, Pearson Education, 2007.
4. B. P.Lathi, "Modern Analog and Digital Communication Systems", 3rd Edition, Oxford University Press, 2007.

Computer Science & Engineering and Information Technology								
Code: IT521	Principles of Programming Languages				L	T	P	C
					3	0	0	3

### Course Outcomes:

1. To introduce the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages.
2. To introduce frameworks for specifying and reasoning about programming languages.
3. Students will be able to summarize different hidden surface elimination algorithms and shading techniques used in computer graphics and digital media production.
4. Students will be able to explain about the technology necessary for creating multimedia content for the web, video, DVD, 2D and 3D graphics, sound and programming.
5. Students can apply the knowledge, techniques, skills and modern tools to become successful professionals in communication and Media industries.

### CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO 1	-	-	2	-	-	-	-	-	-	-	-	-
CO 2	2	2	-	-	-	-	-	-	-	-	-	-
CO 3	2	-	1	-	3	-	-	-	-	-	-	-
CO 4	2	2	-	2	-	1	-	-	-	-	-	-
CO 5	1	2	-	-	-	-	-	-	-	-	-	-

\*3: high, 2: moderate, 1 low

### MODULE 1:

Concepts of structural program development; concept of data types; precedence and associativity of operators; conditional transfer; deterministic and in-deterministic loops

### MODULE 2:

Recursions; functions and procedures - call by value, call by reference and their differences; programming for numerical methods; records.

### MODULE 3:

Data-type handling and various constructs (conditional, loop, functions etc); pointers: concept of pointers and passing parameters using pointers, non-numeric processing, concept of arrays of pointers and pointers to pointers;

**MODULE 4:**

Structures and unions – advantage of using structures, concept of information hiding, pointers to structures; files - basic concept of various types of file access methods: sequential, indexed sequential, random, various statements for file handling

**MODULE 5:**

Advanced Programming Languages like C++, ADA, LISP, PROLOG, and PASCAL. Comparison of various languages

**TEXTBOOK / REFERENCE BOOKS:**

1. Programming Language Pragmatics. Michael Scott, Morgan Kaufmann, 2000.
2. Essentials of Programming Languages. Friedman, Wand and Haynes, Prentice-Hall International (PHI), 1998.
3. Principles of Programming Languages. Tennant. PHI, 1981.

<b>Computer Science &amp; Engineering and Information Technology</b>					
<b>Code: IT522</b>	<b>Semantic Web</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

### **COURSE OUTCOMES:**

1. Understand and explain the overall architecture of semantic web and to illustrate the overview of design principles and technologies in semantic web.
2. Design and implement a small ontology that is semantically descriptive of your chosen problem domain, implement applications that can access, use and manipulate the ontology, represent data from a chosen problem in XML with appropriate semantic tags obtained or derived from the ontology.
3. Describe the semantic relationships among these data elements using Resource Description Framework (RDF).
4. Design and implement a web services application that —discovers the data and/or other web services via the semantic web (which includes the RDF, data elements in properly tagged XML, and the ontology), discover the capabilities and limitations of semantic web technology for different applications.

### **MODULE I: INTRODUCTION**

Introduction to the Syntactic Web and Semantic Web – Evolution of the Web – the Visual and Syntactic Web – Levels of Semantics – Metadata for Web Information – the Semantic Web Architecture and Technologies –Contrasting Semantic with Conventional Technologies– Semantic Modeling -Potential of Semantic Web Solutions and Challenges of Adoption Design Principles.

### **MODULE 2: KNOWLEDGE REPRESENTATION AND ONTOLOGIES**

Knowledge Representation and Reasoning - Ontologies- Taxonomies –Topic Maps – Classifying Ontologies - Terminological Aspects: Concepts, Terms, Relations Between Them – Complex Objects - Subclasses and Sub-properties definitions –Upper Ontologies – Quality – Uses - Types of Terminological Resources for Ontology Building – Methods and Methodologies for Building Ontologies – Multilingual Ontologies -Ontology Development Process and Life Cycle – Methods for Ontology Learning – Ontology Evolution – Versioning Ontologies in Semantic Web.

### **MODULE 3: STRUCTURING AND DESCRIBING WEB RESOURCES**

Structured Web Documents - XML – Structuring – Namespaces – Addressing – Querying – Processing - RDF – RDF Data Model – Serialization Formats- RDF Vocabulary –Inferencing RDFS – basic Idea – Classes – Properties- Utility Properties – RDFS Modelling for Combinations and Patterns- Transitivity.

### **MODULE 4: WEB ONTOLOGY LANGUAGE**

OWL – Sub-Languages – Basic Notions -Classes- Defining and Using Properties – Domain and Range – Describing Properties - Data Types – Counting and Sets- Negative Property Assertions – Advanced Class Description – Equivalence – OWL Logic.

## **MODULE 5: SEMANTIC WEB TOOLS AND APPLICATIONS**

State - of- the- Art in Semantic Web Community-Development Tools for Semantic Web – Jena Framework – SPARL –Querying Semantic Web- Semantic Desktop – Semantic Wikis - Semantic Web Services – Application in Science – Business

### **TEXTBOOKS:**

1. Liyang Yu, |A Developer's Guide to the Semantic Web|, Springer, First Edition, 2011.
2. John Hebel, Matthew Fisher, Ryan Blace and Andrew Perez-opez, —Semantic Web Programming|, First Edition, Wiley, 2009.
3. Grigoris Antoniou, Frank van Harmelen, —A Semantic Web Primer|, Second Edition, MIT Press, 2008.
4. Robert M. Colomb, |Ontology and the Semantic Web|, Frontiers in Artificial Intelligence and Applications, IOS Press, 2007.
5. Dean Allemang and James Hendler, |Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL|, Second Edition, Morgan Kaufmann, 2011.
6. Pascal Hitzler, Markus Krotzsch, Sebastian Rudolph, —Foundations of Semantic Web Technologies, CRC Press, 2009.

### **REFERENCES:**

1. Michael C. Daconta, Leo J. Obrst and Kevin T. Smith, —The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management|, First Edition, Wiley, 2003
2. Karin Breitman, Marco Antonio Casanova and Walt Truszkowski, —Semantic Web: Concepts, Technologies and Applications (NASAMonographs in Systems and Software Engineering) Springer, 2010.
3. Vipul Kashyap, Christoph Bussler and Matthew Moran, The Semantic Web: Semantics for Data and Services on the Web (Data-Centric Systems and Applications), Springer, 2008.



Computer Science & Engineering and Information Technology					
Code: CS531	Web Technology	L	T	P	C
		3	0	0	3

### Course Outcomes:

After Successful completion of course, the students will be able to

1. **Describe** various web technology and application development issues and trends.
2. **Design** static and dynamic web pages using HTML, CSS and Java Script
3. **Design** and implement web services from the server and client side.
4. **Build** interactive web applications using JSP and Servlet.
5. **Identify** the engineering structural design of XML and parse construction tree model.

### CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
<b>CO 1</b>	-	3	-	-	-	-	-	-	-	2	-	-
<b>CO 2</b>	3	2	3	2	3	1	-	-	-	-	-	-
<b>CO 3</b>	-	-	3	-	2		-	-	2	-	-	-
<b>CO 4</b>	2	2	3	-	2	1	-	-	-	-	-	-
<b>CO 5</b>	2	2	-	-	-	-	-	-	-	-	-	-
<b>Avg</b>	2.33	2.25	3	2	2.33	1			2	2		

**Note-** 3: high, 2: moderate, 1 low

### MODULE- 1: INTRODUCTION TO HTML AND WEB PAGES

Introduction to html: Fundamentals of HTML elements, Document body, Different tags, sections, text, hyperlink, lists, tables, color and images, frames, frameset, form.

Web Pages: types and issues, tiers; comparisons of Microsoft and java technologies; WWW: Basic concept, web client and web server, HTTP protocol (frame format), universal resource locator (URL).

### MODULE- 2: DYNAMIC AND ACTIVE WEB PAGES

Dynamic web pages: The need of dynamic web pages; an overview of DHTML, Cascading Style Sheets (CSS), comparative studies of different technologies of dynamic page creation.

Active web pages: Need of active web pages; java applet life cycle.

### MODULE- 3: JAVASCRIPT AND JAVA SERVLET

JavaScript: Data types, variables, operators, conditional statements, array object, date object, string object. Java Servlet: Servlet environment and role, HTML support, Servlet API, the Servlet Life cycle, cookies and sessions.

## **MODULE- 4: JSP**

JSP: JSP architecture, JSP servers, JSP tags, understanding the layout in JSP, Declaring Variables, methods in JSP, inserting java expressions in JSP, processing request from user and generating dynamic response for the user, inserting applets and java beans into JSP, using include and forward action, comparing JSP and CGI program, comparing JSP and ASP program; Creating ODBC data source name, introduction to JDBC, prepare statement and callable statement.

## **MODULE- 5: J2EE AND XML**

J2EE: An overview of J2EE web services, basics of Enterprise Java Beans, EJB vs. Java Beans, basic of RMI, JNI. XML: Basics XML, elements and attributes, document type definition, xml parsers, sequential and tree approach

### **TEXT BOOKS:**

- Chris Bates, “Web Programming: Building Internet Applications”, Wiley Dream Tech, 2<sup>nd</sup> Edition, 2002.
- Jeffrey C K Jackson, “Web Technologies”, Pearson Education, 1<sup>st</sup> Edition, 2006.
- Jason Hunter, William Crawford —Java Servlet Programming| O’Reilly Publications, 2<sup>nd</sup> Edition, 2001.

### **REFERENCE BOOKS:**

- W Hans Bergsten, “Java Server Pages”, O’Reilly, 3<sup>rd</sup> Edition, 2003.
- D. Flanagan, “Java Script”, O’Reilly, 6<sup>th</sup> Edition, 2011.
- Jon Duckett, “Beginning Web Programming”, WROX, 2<sup>nd</sup> Edition, 2008.
- Herbert Schildt, “Java the Complete Reference”, Hill - Osborne, 8<sup>th</sup> Edition, 2011.

### **LIST OF OPEN SOURCE SOFTWARE/LEARNING WEBSITE:**

- Browsers like IE, Mozilla, Firefox etc.
- Server software XAMPP/WAMP/LAMP.
- [www.apachefriends.org](http://www.apachefriends.org)
- [www.w3.org](http://www.w3.org)
- [www.w3schools.com](http://www.w3schools.com)
- [www.php.net](http://www.php.net)
- [www.mysql.com](http://www.mysql.com)
- [www.phpmyadmin.net](http://www.phpmyadmin.net)
- [www.javatpoint.com](http://www.javatpoint.com)

Computer Science & Engineering and Information Technology					
Code: CS532	Computer Architecture*	L	T	P	C
		3	0	0	3

\*This course open to all branch except CSE/IT.

#### Course Outcomes:

1. Ability to describe the organization of computer and machine instructions and programs
2. Ability to analyze Input / Output Organization
3. Analyze the working of the memory system and basic processing unit.
4. Ability to solve problems of multicores, multiprocessors and clusters.
5. Choose optical storage media suitable for multimedia applications.

#### CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	-	3	-	2	2	-	-	-	-	-	-	1
CO2	2	2	2	2	2	-	-	-	-	-	-	2
CO3	2	2	2	2	3	-	-	-	-	-	-	2
CO4	3	3	3	2	2	-	-	-	-	-	-	2
Average												

\*3: high, 2: moderate, 1 low

#### MODULE 1:

**Basics of Digital Electronics:** Multiplexers and De multiplexers, Decoder and Encoder, Codes, Logic gates, Flip flops, Registers.

**Register Transfer and Micro Operations:** Bus and Memory Transfer, Logic Micro Operations, Shift Micro Operations, Register transfer and register transfer language, Design of arithmetic logic unit.

#### MODULE 2:

**Basic Computer Organization:** Instruction codes, Computer instructions, Timing and Control, Instruction cycle, Memory reference Instruction, Complete computer description, Design of basic computer, Input output and interrupt.

#### MODULE 3:

**Control Unit:** Hardwired controls, Micro programmed controls.

**Central Processing Unit :** Program control, Reduced instruction set computer, Complex instruction set computer, Data Transfer, Manipulation, General register and stack organization, Addressing mode.

#### MODULE 4:

**Computer Arithmetic:** Addition and subtraction algorithm, Multiplication algorithm, Division algorithms.

#### MODULE 5:

**Input-Output Organization:** Priority interrupt, Peripheral devices, Input output interface, Data transfer schemes, Program control and interrupts, Direct memory access transfer, Input/output processor.

**Memory Unit:** High speed memories, Memory hierarchy, Processor Vs Memory speed, Cache memory, Associative memory, Inter leave, Virtual memory, Memory management.

**MODULE 6:**

**Introduction to Parallel Processing:** Pipelining, Characteristics of multiprocessors, Interconnection structures, Inter processor arbitration, Inter processor communication, Synchronization.

**Text Books:**

1. Computer System Architecture by Morris Mano, Prentice hall, 3<sup>rd</sup> Edition, (2007)

**References:**

1. Computer Organization by Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Tata Mcgraw Hill, 5th Edition, (2011)
2. Computer Architecture : A Quantitative Approach by Hennessy, J. L, David A Patterson, and Goldberg, Pearson Education, 4<sup>th</sup> Edition, (2006)

<b>Computer Science &amp; Engineering and Information Technology</b>					
<b>Code: CS533</b>	<b>Data Structures and Algorithms*</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

\*This course is open to all branches except CSE/IT.

### **COURSE OUTCOMES:**

After Completion of this course, the students will be able to:

1. Understand various types of fundamental data structures (standard and user defined).
2. Determine and analyze about algorithm for the run time complexities and the space requirements.
3. Acquire knowledge of data structures and algorithms for implementing various computing system

### **MODULE 1**

Linear Data Structures - Sequential representations - Arrays and Lists, Stacks, Queues and Dequeues, strings, Application, Linear Data Structures, Link Representation - Linear linked lists, Circularly linked lists. Doubly linked lists, application.

### **MODULE 2**

Non-linear Data Structure: Trees - Binary Trees, Traversals and Threads, Binary Search Trees, Insertion and Deletion algorithms, Height-balanced and weight-balanced trees, B-trees, B+ -trees, Application of trees; Recursion - Design of recursive algorithms, Tail Recursion, When not to use recursion, Removal of recursion.

### **MODULE 3**

Graphs - Representations of graphs (adjacency list, adjacency matrix, Sparse Matrix); Breadth-first and Depth-first Search. Hashing: Hash tables, including collision-avoidance strategies;

### **MODULE 4**

Asymptotic analysis of upper and average complexity bounds; Identifying differences among best, average, and worst case behaviors; Big O, little o, omega, and theta notation; Time and space tradeoffs in algorithms;

### **MODULE 5**

Sorting and Searching Algorithms- Sequential and binary search algorithms; Bubble sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort and Radix Sort. Brute-force algorithms; Greedy algorithms; Divide-and-conquer; Backtracking; Shortest path algorithms (Single source shortest path; Dijkstra's and Floyd's algorithms); Minimum spanning tree (Prim's and Kruskal's algorithms);

### **TEXT / REFERENCE BOOKS**

1. S. Sahni, Data structures, algorithms, & applications in Java, McGraw-Hill, 2005
2. J. P. Trembly et al, An introduction to data structures with applications, McGraw- Hill, 2007
3. D. E. Knuth, Art of computer programming, Volume 1: Fundamental algorithms, Addison-Wesley, 2011
4. Thomas H. Cormen , Charles E. Leiserson , Ronald L. Rivest , Clifford Stein, Introduction to Algorithms, 3rd Edition, PHI, 2009

Computer Science & Engineering and Information Technology					
Code: CS551	Operating System Lab	L	T	P	C
		0	0	3	1

### **COURSE OUTCOMES:**

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

1. Appreciate the advantages of Unix OS.
2. Develop and debug, C programs created on UNIX platforms.
3. Use and if necessary install standard libraries.

### **LIST OF EXPERIMENTS:**

1. Study of Basic commands of Linux.
2. Write a Shell script to print given numbers sum of all digits
3. Write a shell script to validate the entered date. (eg. Date format is: dd-mm-yyyy)
4. Write a shell script to check entered string is palindrome or not.
5. Write a Shell script to say Good morning/Afternoon/Evening as you log in to system.
6. Write a C program to create a child process
7. Finding out biggest number from given three numbers supplied as command line arguments
8. Printing the patterns using for loop.
9. Shell script to determine whether given file exist or not.
10. Write a program for process creation using C. (Use of gcc compiler).
11. Implementation of FCFS Algorithm.
12. Implementation of Round Robin Algorithm.
13. Implementation of Banker's Algorithm.

Computer Science & Engineering and Information Technology					
<b>Code: CS552</b>	<b>Database Management System Lab</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>0</b>	<b>0</b>	<b>3</b>	<b>1</b>

#### H/W Requirement:

- At least Dual Core or Core-I3 Pro Computing System, 2GB RAM, 80GB HDD

#### Software

- All systems are configured in DUAL BOOT mode i.e., Students can boot from Windows 7/8 or Linux as per their lab requirement.
- MySQL/PgSQL/Oracle May be used as Database Management System
- Systems are provided for students in the 1:1 ratio.

#### COURSE OUTCOMES

1. Design and implement a database schema for a given problem-domain
2. Create and maintain tables using PL/SQL
3. Populate and query a database Course
4. Application development using PL/SQL & front end tools

#### LIST OF EXPERIMENTS

- 1 Database Schema for a customer-sale scenario  
Customer(Cust id : integer, cust\_name: string)  
Item(item\_id: integer, item\_name: string,  
price: integer)  
Sale(bill\_no: integer, bill\_data: date, cust\_id: integer, item\_id: integer,  
qty\_sold: integer) For the above schema, perform the following—
  - a) Create the tables with the appropriate integrity constraints
  - b) Insert around 10 records in each of the tables
  - c) List all the bills for the current date with the customer names and item numbers
  - d) List the total Bill details with the quantity sold, price of the item and the final amount
  - e) List the details of the customer who have bought a product which has a price>200
  - f) Give a count of how many products have been bought by each customer
  - g) Give a list of products bought by a customer having cust\_id as 5
  - h) List the item details which are sold as of today
  - i) Create a view which lists out the bill\_no, bill\_date, cust\_id, item\_id, price, qty\_sold, amount
  - j) Create a view which lists the daily sales date wise for the last one week
- 2 Database Schema for a Student Library scenario  
Student(Stud\_no : integer, Stud\_name: string)  
Membership(Mem\_no: integer, Stud\_no:  
integer) Book(book\_no: integer,  
book\_name:string, author: string)  
Iss\_rec(iss\_no:integer, iss\_date: date, Mem\_no: integer,  
book\_no: integer) For the above schema, perform the following—



- a) Create the tables with the appropriate integrity constraints
- b) Insert around 10 records in each of the tables
- c) List all the student names with their membership numbers
  
- d) List all the issues for the current date with student and Book names
- e) List the details of students who borrowed book whose author is CJDATE
- f) Give a count of how many books have been bought by each student
- g) Give a list of books taken by student with stud\_no as 5
  
- h) List the book details which are issued as of today
- i) Create a view which lists out the iss\_no, iss\_date, stud\_name, book name
- j) Create a view which lists the daily issues-date wise for the last one week

3 Database Schema for a Employee-pay

scenario employee(emp\_id : integer,

emp\_name: string) department(dept\_id:

integer,dept\_name:string)

paydetails(emp\_id : integer, dept\_id: integer, basic: integer, deductions: integer, additions: integer, DOJ: date)

payroll(emp\_id : integer, pay\_date: date)

For the above schema, perform the following—

- a) Create the tables with the appropriate integrity constraints
- b) Insert around 10 records in each of the tables
- c) List the employee details department wise
  
- d) List all the employee names who joined after particular date
- e) List the details of employees whose basic salary is between 10,000 and 20,000
- f) Give a count of how many employees are working in each department
- g) Give a names of the employees whose netsalary>10,000
  
- h) List the details for an employee\_id=5
- i) Create a view which lists out the emp\_name, department, basic, dedeuctions, netsalary
- j) Create a view which lists the emp\_name and his netsalary

4 Database Schema for a Video Library scenario

Customer(cust\_no: integer,cust\_name: string)

Membership(Mem\_no: integer, cust\_no: integer)

Cassette(cass\_no:integer, cass\_name:string,

Language: String)

Iss\_rec(iss\_no: integer, iss\_date: date, mem\_no: integer, cass\_no: integer)

For the above schema, perform the following—

- a) Create the tables with the appropriate integrity constraints
- b) Insert around 10 records in each of the tables
  
- c) List all the customer names with their membership numbers
- d) List all the issues for the current date with the customer names and cassette names
- e) List the details of the customer who has borrowed the cassette whose title is “ The Legend”
- f) Give a count of how many cassettes have been borrowed by each customer
  
- g) Give a list of book which has been taken by the student with mem\_no as 5
- h) List the cassettes issues for today
- i) Create a view which lists out the iss\_no, iss\_date, cust\_name, cass\_name
- j) Create a view which lists issues-date wise for the last one week

5 Database Schema for a student-Lab scenario

Student(stud\_no: integer, stud\_name: string, class: string)

Class(class: string, descrip: string)

Lab(mach\_no: integer, Lab\_no: integer, description: String)

Allotment(Stud\_no: Integer, mach\_no: integer, dayof week: string)

For the above schema, perform the following—

- a) Create the tables with the appropriate integrity constraints
- b) Insert around 10 records in each of the tables
- c) List all the machine allotments with the student names, lab and machine numbers
- d) List the total number of lab allotments day wise
  
- e) Give a count of how many machines have been allocated to the ‘CSIT’ class
- f) Give a machine allotment details of the stud\_no 5 with his personal and class details
- g) Count for how many machines have been allocated in Lab\_no 1 for the day of the week as “Monday”
- h) How many students class wise have allocated machines in the labs
  
- i) Create a view which lists out the stud\_no, stud\_name, mach\_no, lab\_no, dayofweek
- j) Create a view which lists the machine allotment details for “Thursday”.

6 a) Write and execute subprogram to find largest number from the given three numbers.

b) Write and execute subprogram using loop, while and for iterative control statement.

7. a) Write and execute subprogram to check whether the given number is Armstrong or not b) Write and execute subprogram to generate all prime numbers below 100.

a) Write and execute subprogram to demonstrate the GOTO statement. 8 b) Write a subprogram to demonstrate %type and %rowtype attributes

9 a) Write and execute subprogram to demonstrate predefined exceptions

b) Write and execute subprogram to demonstrate user defined exceptions

10 a) Create a cursor, which displays all employee numbers and names from the EMP table.

b) Create a cursor, which update the salaries of all employees as per the given data.

11 a) Create a cursor, which displays names of employees having salary > 50000.

b) Create a procedure to find reverse of a given number

- 12 a) Create a procedure to update the salaries of all employees as per the given data  
b) Create a procedure to demonstrate IN, OUT and INOUT parameters
- 13 a) Create a function to check whether given string is palindrome or not.  
b) Create a function to find sum of salaries of all employees working in department number 10.
- 14 a) Create a trigger before/after update on employee table for each row/statement.  
b) Create a trigger before/after delete on employee table for each row/statement.  
c) Create a trigger before/after insert on employee table for each row/statement.
- 16 a) Create a Form to display employee details using SQL  
  
b) Create a Report to generate all employee annual salaries....

#### **Additional Programs**

- 1 **Create a form using Forms 6i to display Employee table data.**
- 2 **Create a Master/details relationship form which perform Add New, Search, Delete, Save and Update on the records**
- 3 **Generate a report to calculate employee's salaries department wise from employee table.**
- 4 **Create a Report to generate the details of employee table including sum and average salaries department wise. Course Outcome: Upon successful completion of this Lab the student will be able to:**
  - **Creating , Modify database Tables and manipulate data in the Tables**
  - **Performing database operations in a procedural manner using pl/sql**
  - **Performing database operations (create, update, modify, retrieve, etc.,) using front-end tools like D2K/Gambas/PHP/Servlet/JSP.**
  - **Design and Develop applications like banking, reservation system, etc.**
  - **Design and implement a small database project using RDBMS to understand the concept of a database transaction and related database facilities.**