FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: CS30106 **REF NO: To be filled by CD office**

Computer Organization

Credits:3 **Teaching Scheme: -** Theory 3Hrs/Week

Prerequisites:

1. Digital Electronic.
2. Computer Programming

Objectives:

1. To get idea about different hardware used in Computer system and their interconnections.
2. To get knowledge on how hardware and software are interlinked to process instruction.
3. To get knowledge on different types of memory, their utility and mapping.
4. To get idea about how control unit control the execution of instructions.
5. To realize how arithmetical operations are performed inside ALU using various registers.
6. To get idea on storage and I/O transfer.

Course Details:

Unit1: Structure of a Computer System (7 Hrs)

U1.1: Organization & Architecture, Structure & Function,Brief History of computers, Von Neumann Architecture, Bus Structure, Elements of Bus Design, CPU Architecture, Register Organization, Instruction types,Types of operands,Instruction formats, addressing modes and address translation. Instruction cycles, RISC and SISC Processors, Pipelining, Pipelining Hazards, Superscalar Processors. Performance consideration..

U1.2Self Study: PCI Bus, Basic multiprocessor architecture.

Unit2: Memory Organization (7 Hrs)

U2.1. Hierarchical memory system, Characteristics, Size, Access time, Read Cycle time and address space. Main Memory Organization, types of memory, memory chip design, Cache memory Organization: Address mapping, Cache Coherence, interleaved memories. Virtual Memory: Paging.

U2.2. **Self Study:** Internal structure of RAM & ROM, Multi Level Cache.

Unit3: Processor and Control Unit (7 Hrs)

U3.1. Fundamental Concepts: Single Bus CPU organization, Register transfers, Performing

an arithmetic/ logic operations, fetching a word from memory, storing a word in memory, Execution of a complete instruction. Micro- operations, Hardwired Control and Micro-programmed Control CU.

U3.2. Self Study: Multiple- bus organization. Applications of micro programming.

Unit4: Computer Arithmetic (7 Hrs)

U4.1. Integer Representation: Fixed point & Signed numbers. Integer Arithmetic: 2's Complement arithmetic, addition, subtraction, multiplication, Booth's Algorithm, Division with restoring algorithm and non restoring algorithm, Floating point representation of number: IEEE754 Standards for Floating point representations (Single Precision Format & Double Precision Format).

U4.1. Self Study: ALU Design.

Unit5: Secondary Storage and I/O (7 Hrs)

U5.1. Magnetic Disk, Read Write mechanism of HDD, Access time, Latency time, Optical memory, CDROM. Basic fundamentals of Input Output organization –Peripheral devices, Asynchronous Data Transfer, Mode of Data Transfer, Direct Memory Access (DMA), Bus Arbitration.

U5.2. Self Study: Input Output Processor (IOP), Interrupts, Vectored Interrupt, Interrupt Handling

Text Books:

T1. "Computer Organization", C. Hamacher, V. Zvonko, S. Zaky, Tata McGraw Hill Publication,ISBN 007-120411-, 5th Edition.
T2. "Computer System Architecture", M. Morris Mano, Pearson Education, ISBN-978-81-317-0070-9, 3rd Edition.

Reference Books:

R1."Computer Architecture and Parallel Processing", Hwang and Briggs, Tata McGraw Hill Publication , ISBN 13: 9780070315563.
R2. "Structured Computer Organization", A. Tanenbaum, Prentice Hall Publication, ISBN 81–203–1553–7, 4th Edition.

Course Outcomes:
Upon completion of the course, graduates will be able –

CO1: Identify and analyze the basic structures of a computer hardware units, connectivity and software.
CO2: Design the basic structure of machine instruction and programs, memory location.
CO3: Analyze different memory in the hierarchy, their mapping and their performance.
CO4: Analyze internal details of a processor, how instructions are executed using different hardware units, and how control unit controls all hardware components.

CO5: Study the design of ALU for arithmetic operations and use of registers.

CO6: Analyze the organization of secondary storage and how all the I/O devices communicate with CPU and transfer data.

COURSE CODE: CS30106 **REF NO: To be filled by CD office**

Computer Organization Tutorial

Credits: 1 **Teaching Scheme: -** Tutorial 1Hr/Week

Prerequisites:

1. Digital Electronic.
2. Computer Programming

Objectives:

1. To practice assembly language programming and measure the influence of implementing pipeline on a processor.
2. Practice different numerical on memories used in a computer and also to realize how address mapping takes place between different memories.
3. To get idea on micro instructions and CU design.
4. Practice how arithmetical operations are performed inside ALU using various registers.
5. Study on different modern multi-core processors.

Tutorial No.1: Assembly language programming

Tutorial No.2: Numerical on throughput, efficiency, speedup of a pipelined processor

Tutorial No.3: Problems on Cache Mapping

Tutorial No.4: Problems on VM mapping using paging.

Tutorial No.5: Design control steps for instructions.

Tutorial No.6: Problems on control unit design.

Tutorial No.7: Fast adder and fast multiplier.

Tutorial No.8: Floating addition, subtraction, multiplication, division.

Tutorial No.9: Numerical on SSD.

Tutorial No.10: Case study on multi core processor.

Course Outcomes: Upon completion of the course, graduates will be able –

CO1: Identify and analyze the basic structures of a computer hardware units, connectivity and software.

CO2: Design the basic structure of machine instruction and programs, memory location.

CO3: Analyze different memory in the hierarchy, their mapping and their performance.

CO4: Analyze internal details of a processor, how instructions are executed using different hardware units, and how control unit controls all hardware components.

CO5: Study the design of ALU for arithmetic operations and use of registers.

CO6: Analyze the organization of secondary storage and how all the I/O devices communicate with CPU and transfer data.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE:CS30306 **REF NO: To be filled by CD office**

Computer Organization Lab

Credits:01 **Teaching Scheme: -** Laboratory 02Hrs/Week

Prerequisites:

1. Digital Electronic.
2. Computer Programming.

Objectives:

3. Details study of each and every hard ware components used in a computer.
4. Study of Interfacing of different hardware.
5. Study of Power supply to the different components of the computer .
6. Assembling of a PC.
7. Idea on VHDL coding.

Course Details:

- Study on mother board and different ports and slots connected to it.
- Study on different storage devices.
- Study on different input and output devices.
- Study on power supply unit and how it provides power supply to different hardware components.
- Fundamental to VHDL coding.

List of Experiments:

Experiment No. 1: To study about the different I/O ports using trainer kit.

Experiment No. 2: To study about chipsets, ports and slots of motherboard using trainer kit.

Experiment No.3: To study on the internal architecture of HDD using trainer kit.

Experiment No.4: To study on internal architecture and function of keyboard using trainer kit.

Experiment No.5: To study on internal architecture and function of mouse using trainer kit.

Experiment No.6: To study on internal architecture and function of the printer using the trainer kit.

Experiment No.7: To study on internal architecture and function of optical disk drive using trainer kit.

Experiment No.8: To study on internal architecture and function of monitor using trainer kit.

Experiment No.9: To study on internal architecture and function of SMPS using the trainer kit.

Experiment No.10: To study on dismantling and assembling of PC.

Experiment No.11: Experiments on simple fundamental units like half adder, full adder, using VHDL code.

Experiment No.12: Multiplexer, De-multiplexer using VHDL code.

Text Books:

T1. Patterson, D.A., and Hennessy, J.L. , "Computer Organization and Design: The Hardware/Software Interface", Morgan Kaufmann Publishers, 4th Edition, Inc.2005

T2. Michael Meyers, Lloyd Jeffries, " PC Hardware", McGraw Hill Professional, Inc. 2004

T2. VHDL Programming by Perry

Reference Books:

R1. "Computer System Architecture", M. Morris Mano, Pearson Education, ISBN-978-81-317-0070-9, 3rd Edition

R2. "Computer Architecture", Nicholas Carter, 2002, T.M.H.

Course Outcome:

CO1: Identify components on a motherboard.

CO2: Experiment on read and write mechanism of HDD and prepare HDD using formatting and partitioning of it.

CO3: Use different I/O devices attached to a computer.

CO4: Analyze and experiment on power supply to a computer.

CO5: Gain hand on experience for assembling a personal computer.

CO6: Apply VHDL coding to realize different digital circuits.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: CS30108 **REF NO: To be filled by CD office**

Theory of Computation

Credits: 3 **Teaching Scheme: -** Theory 3 Hrs/Week

Prerequisites:

1. Computer Programming
2. Discrete mathematics

Objectives:

1. To introduce mathematical and computational principles of programming languages.
2. To study and design automata for different class of problems.

Course Details:

Unit 1 **(6 Hrs)**

Title- Introduction to Automata Theory

U1.1

Concepts to Automata Theory: Alphabets, Strings, Languages and Grammar Deterministic finite Automata (DFA) and Nondeterministic finite Automata (NFA), NFA with epsilon transition, Equivalence of NFA and DFA, Minimization of Automata, Conversion of NFA with epsilon to DFA Equivalence, Chomsky Classification.

U1.2

Self Study: Concepts to Automata with outputs Moore and Mealy Machine.

Unit II **(6 Hrs)**

U2.1

Regular Expression and Languages: Definition, Identities, Arden's theorem, Kleen's Theorem, Regular expression to DFA, DFA to Regular expression, Non Regular Languages, Pumping Lemma for regular Languages, Closure properties of Regular Languages.

U2.2

Self Study: Conversion from Automata to Grammar and vice versa

Unit III **(6 Hrs)**

Title- Context Free Grammars (CFG) and Push Down Automata (PDA)

U3.1

Context Free Grammars: Definition of CFG, Parse trees, Ambiguity in Grammar, Ambiguous and Unambiguous CFG, Inherent ambiguity, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure properties of CFLs, Cook Younger Kasami(CYK) algorithm and Chomsky Hierarchy. Pumping Lemma for CFLs. Definition and Acceptability of PDA, Language to PDA and CFG to PDA

U3.2

Self Study: Ogden's lemma and Parikh's theorem, Early's algorithm.

Unit IV **(6 Hrs)**

Title- Turing Machines, Un-decidability & Computable function

U4.1

Turing Machines: Definition and representation of TM, Language acceptance by TM. Variants of TM, Universal Turing Machines, Godel numbering, Church-Turing Thesis,

Recursive and Recursively Enumerable languages. Halting problem, Post Correspondence Problem, Introduction to countable and uncountable sets, Recursive function, Primitive recursive function, Ackerman's function.

U4.2

Self Study: Linear Bounded Automata and Contest sensitive language and Modified PCP

Unit V **(6 Hrs)**

Title- Time Complexity

U5.1

Class P, class NP, NP-Completeness and Reducibility

U5.2

Self Study: Proofs on Class P, NP and NP-C.

Text Books

T1. "Introduction to Automata Theory, Languages and Computation", Hopcroft J, Motwani R, Ullman ,Addison-Wesley, ISBN 81-7808-347-7, Second Edition .

T2. "Introduction to Theory of Computation", Michael Sipser, Course Technology, ISBN-10: 053494728X, Third Edition.

Reference Books

R1. Introduction to Formal Languages, Automata Theory and Computation: K. Kirthivasan, Rama R, Pearson Education.

R2. Theory of computer Science (Automata Language & computations) K.L. Mishra N. Chandrashekhar, PHI.

R3. Introduction to Languages and the Theory of Computation, J. Martin, Tata McGraw-Hill, ISBN 0-07-049939-x, Third edition, 2003.

R4. Elements of The theory of Computation, H.R.Lewis, C.H.Papadimitriou, Pearson Education, ISBN 81-7808-487-2, Second Edition.

R5. Introduction to Languages and the Theory of Computation: Martin, Tata McGraw Hill, 3rd Edition.

R6. Formal Languages and Automata Theory, C.K.Nagapal, Oxford University Press, First Edition, 2011.

Course Outcomes:

Upon completion of the course, graduates will be able to –

CO1: Prove results using proof by induction, proof by contradiction, proof by construction.

CO2: Describe various automata theoretic models for recognizing formal languages and transform regular expressions and grammars.

CO3: Distinguish different computing languages and classify their respective types.

CO4: Construct pushdown automata and the equivalent context free grammars and prove the equivalence of the languages described by pushdown automata and context free grammars.

CO5: Design Turing Machine and prove the equivalence of the languages described by Turing machines and Post machines.

CO6: Analyze algorithmic complexity, computability and solvability of problems.

COURSE CODE: CS30108 **REF NO: To be filled by CD office**

Theory of Computation Tutorial

Credits: 01 **Teaching Scheme:** Tutorial 01 Hrs/Week

Prerequisites:

3. Computer Programming
4. Discrete mathematics

Objectives:

3. To introduce mathematical and computational principles of programming langauage.
4. To study and design automata for different class of problems.

List of Tutorials

Tutorial No. 1: Problems on NFA to DFA conversion.

Tutorial No. 2: Numerical based on minimization and equivalence of Automata.

Tutorial No. 3: Exercises on conversion of Regular expression to DFA and vice versa

Tutorial No. 4: Problems on proof of Closure properties of Regular Languages.

Tutorial No. 5: Problems on proof of whether a grammar is regular or not by using pumping lemma.

Tutorial No. 6: Problems on checking of Ambiguity of Grammar and Simplification of CFGs,

Tutorial No. 7: Problems on Normal forms of CFGs: CNF and GNF.

Tutorial No. 8: Problems based on CFG to PDA construction.

Tutorial No. 9: Problems on Turing machine.

Tutorial No. 10: Problems on Ackermann's function, Gödel Numbering, Post Correspondence Problem

Text Books

T1. "Introduction to Automata Theory, Languages and Computation", Hopcroft J, Motwani R, Ullman ,Addison-Wesley, ISBN 81-7808-347-7, Second Edition .
T2. "Introduction to Theory of Computation", Michael Sipser, Course Technology, ISBN-10: 053494728X, Third Edition.

Reference Books

R1. Introduction to Formal Languages, Automata Theory and Computation: K. Kirthivasan, Rama R, Pearson Education.
R2. Theory of computer Science (Automata Language & computations) K.L. Mishra N. Chandrashekhar, PHI.
R3. Introduction to Languages and the Theory of Computation, J. Martin, Tata McGraw-Hill, ISBN 0-07-049939-x, Third edition, 2003.
R4. Elements of The theory of Computation, H.R.Lewis, C.H.Papadimitriou, Pearson Education, ISBN 81-7808-487-2, Second Edition.
R5. Introduction to Languages and the Theory of Computation: Martin, Tata McGraw Hill, 3rd Edition.
R6. Formal Languages and Automata Theory, C.K.Nagapal, Oxford University Press, First Edition, 2011.

Course Outcomes:
Upon completion of the course, graduates will be able –

CO1: The ability to prove results using proof by induction, proof by contradiction, proof by construction.
CO2: Ability to describe various automata theoretic models for recognizing formal languages and transform regular expressions and grammars.
CO3: Distinguish different computing languages and classify their respective types.
CO4: Able to construct pushdown automata and the equivalent context free grammars and prove the equivalence of the languages described by pushdown automata and context free grammars.
CO5: Able to design Turing Machine and prove the equivalence of the languages described by Turing machines and Post machines.
CO6: Analyze algorithmic complexity, computability and solvability of problems.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: IT30104 **REF NO: To be filled by CD office**

Concept of Data Modeling

Credits: 3 **Teaching Scheme: - 3 Hrs/Week**

Prerequisites:
1. Computer Programming
2. Knowledge of Data Structures

Objectives:

To introduce database systems and learn the techniques of data modeling, database design, data retrieval and database management.

Course Details:

Unit 1 **(7 Hrs)**

Title Introduction to DBMS and ER Data Model

U1.1

Data Storage: File processing system, Disadvantages; DBMS: Need of DBMS, Terms: Data, Database, Metadata, Data Dictionary, Database System, Database Management System, Data Abstraction, Data Independence, System Architecture of DBMS, Data Model: Definition, ER and Relational Data Model, Comparison of Different Data Models, ER Model: Entity, Entity Set, Attributes, Primary Key, Relationship, Types and Attributes of Relationship, Role, Cardinality Ratio, Participation Constraint, Weak Entity Set, EER Features.

U1.2

Self-Study: Hierarchical and Network Data Models, Tools for Designing ER Model

Unit 2 **(7 Hrs)**

Title - Relational Data Model and Relational Algebra, Relational Calculus

U2.1

Relational Data Model: Terms: Relation, Schema, Attributes, Tuples, Domains, Relation Degree (or Arity) and Cardinality, , Super Key, Candidate Key, Primary Key and Foreign Key, Relational Model Constraints, Schema Diagram, ER to Relation Mapping, Relational Algebra and its Operations: Set Theoretic Operators (Union, Intersection, Cartesian product, Division), Relational Algebra operators (Projection, Selection, Join, Rename.

Relational Calculus: TRC, DRC

Database Language: SQL (DDL, DML, DCL), QBE

U2.2

Self Study: Characteristics of Relation, Codd's Twelve Rules for Relational DBMS, Reverse Engineering: Relational Database into ER/ EER Model. Case study Using PL/SQL, DB functions (Date, Timestamp), Cursors.

Unit 3 (8 Hrs)

Title- Normalization

U3.1

Normalization: Anomalies of un-Normalized Relation, Need of Normalization, Pros and Cons of Normalization, Functional Dependency: Trivial, Full, Partial, Transitive, Multivalued, Join, Inclusion Dependency, Dependency Diagram, Inference Rules for Functional Dependencies, Closure of Functional Dependencies, Algorithms to find: 1. Candidate Key, 2. Closure of Attribute Set, 3. Minimal Cover of Functional Dependencies, Normal Forms: Checking of Lossless Join Decomposition and Dependency Preservation, Normal Forms: 1NF, 2NF, 3NF, BCNF, 4NF.

U3.2

Self-study :Normal Forms: 5NF and DKNF, Normalization at Conceptual Level.

Unit 4 (7 Hours)

Title- Transaction Management

U4.1

Complexity Theory: Transaction: Concept, ACID properties, Transaction States; Schedule: Definition, Types of Schedule, Serializability, Conflict and View Serializability, Precedence Graph, Recoverable Schedule, Cascade less Schedule, Deadlock, Concurrency Control Protocols: Lock Based, Timestamp Based Protocol

U4.2

Self-study: Recovery System: Log based Recovery, Checkpoint, Shadow paging. Deadlock Handling.

Unit 5 (7 Hours)

Title- Object Oriented and Object relational Database Systems

U5.1

Object relational database systems: Extensibility features and objects orientation in relational database systems, ODBC, JDBC, XML/Web databases: semi-structured data, querying, Database system architectures (2-Tierand 3-Tier), And Client-servers architecture. Parallel and distributed database architectures, Performance issues.

U5.2

Self-Study: Object orientation in relational systems, Data mining, Data warehousing, Online analytical processing Structured and Unstructured data

Note: Five assignments to be given to the students on self-study, comprising of one assignment from each unit.

Text Books:

1 Database System Concepts, Silberschatz, Korth, Sudarshan, McGraw Hill International Edition, ISBN- 0-07-228363-7, 4th Edition

2. "Fundamentals of Database Systems", Elmasri and Navathe, Pearson Education, ISBN 81-297-0228-2, 4th Edition.

References Books:

1. An introduction to Database System – Bipin Desai, Galgotia Publications
2. Database System: concept, Design & Application by S.K.Singh (Pearson Education)
3. Database Modeling and Design: Logical Design by Toby J. Teorey, Sam S.Lightstone, and Tom Nadeau, "", 4th Edition, 2005, Elsevier India Publications, New Delhi
4. Fundamentals of Database Management System – Gillenson, Wiley India

Course Outcomes:

Upon completion of the course, graduates will be able to –

1. Apply the basic concepts of DBMS to maintain the database and protect it.
2. Use different Data model concepts to design the appropriate database according to the requirement.
3. Use different design techniques to design the database (Relational).
4. Apply the DDLC and guidelines to avoid redundancy and anomalies.
5. Use the tools to connect frontend and backend.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: CS30109 **REF NO: To be filled by CD office**

Operating System

Credits: 3 **Teaching Scheme: -** Theory 3 Hrs/Week

Prerequisites:

1. Computer Organization
2. Computer Programming

Objectives:

1. To understand main components of OS and their working
2. To study the operations performed by OS as a resource manager
3. To understand the different scheduling policies of OS
4. To understand the different memory management techniques
5. To understand process concurrency and synchronization
6. To understand the concepts of input/ output, storage and file management
7. To study different OS and compare their features.

Course Details:

Unit 1

Title- Introduction to OS **(6hrs)**

U1.1.

Operating Systems Objectives and functions, Components of OS, OS Structure, Evolution of Operating Systems - Simple Batch, Multiprogrammed, time shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, Operating System services, System Calls.

U1.2. Self Study: System Programs, System structure, Virtual Machines, Dual Mode Operation.

Unit 2

Title – Study of Process Management: **(6Hrs)**

U2.1.

Process Management:

Process and CPU Scheduling - Process concepts – Process and Process States, Process Control Block, Cooperating Processes, Inter-process Communication, Process Scheduling - Scheduling Queues, Schedulers, Context Switch, Non Pre-emptive and Pre-emptive Scheduling, Dispatcher, Schedulability Criteria, Scheduling algorithms.

Process Coordination: Process Synchronization, The Critical section Problem, Peterson's solution, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Monitors. Threads and its type.

U2.2. Self Study: Process creation mechanism and scheduling algorithms used in Linux and Windows.

Unit 3
Title – Deadlock: (6 Hrs)

U3.1
Deadlocks: System model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock avoidance, Deadlock Detection, recovery from Deadlock**.**
U3.2. Self Study: Thread creation and Thread scheduling in Linux and Windows.

Unit 4
Title – Memory Management: (6 Hrs)

U4.1
Memory Management strategies, Background, Logical versus Physical Address space, MMU, Address Translation, Swapping, Contiguous Allocation, Paging, Segmentation.
Virtual Memory: Background, Demand paging, Page Replacement concepts, Page Replacement Algorithms. Allocation of frames, Thrashing, Segmentation with Paging,
U4.2. Self Study: Multilevel Paging, Inverted Page Table, Demand Segmentation, Case study using Linux and Windows.

Unit 5
Title- File System & Storage Management (8 Hrs)

U5.1.
File System Interface - The Concept of a File, Access methods, Directory Structure, File System Mounting, File Sharing, Protection, File System Structure, File System Implementation, Allocation methods.
I/O Systems- Overview of Mass Storage Structure, Device Drivers, Disk Structure, Disk Scheduling, Disk Management, and Swap space Management, Free-space Management, Directory Implementation, RAID Structure.
U5.2. Self Study: Disk Attachment, Stable Storage Implementation, Case studies on File system: LINUX and Windows.

Note: Five assignments to be given to the students on self study, comprising of one assignment from each unit.

Text Books:
1. Operating System Principles, Abraham Silberchatz, Peter B. Galvin, Greg Gagne 9th Edition, Wiley Student Edition.
2. Operating systems - Internals and Design Principles, W. Stallings, 6th Edition, Pearson.

Reference Books:

1. Modern Operating Systems, Andrew S Tanenbaum 3rd Edition PHI.
2. Operating Systems A concept - based Approach, 2nd Edition, D. M. Dhamdhere, TMH.
3. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
4. Operating Systems, A. S. Godbole, 2nd Edition, TMH
5. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
6. Operating Systems, S, Haldar and A. A. Arvind, Pearson Education.
7. Operating Systems, R. Elmasri, A. G. Carrick and D. Levine, Mc Graw Hill.
8. Operating Systems in depth, T. W. Doeppner, Wiley.

Course Outcome

1. Compare and contrast different types of operating system and analyze the fundamentals of operating system concepts, services, and components.
2. Analyze and understand operating system process scheduling criteria, scheduling algorithms, threading issues, thread scheduling and inter-process communication mechanism.
3. Analyze process synchronization mechanism by studying and using classical problems of synchronization.
4. Characterize methods and mechanisms to handle, prevent, avoid and detect deadlock in a system and inspect main memory and virtual memory management strategies.
5. Analyze file and directory structure management along with I/O management issues.
6. Carry out case studies in different contemporary operating systems.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: CS30109 **REF NO: To be filled by CD office**

Operating System Tutorial

Credits: 01 **Teaching Scheme:** Tutorial 01 Hrs/Week

Prerequisites:

3. Computer Organization
4. Computer Programming

Objectives:

8. To understand main components of OS and their working
9. To study the operations performed by OS as a resource manager
10. To understand the different scheduling policies of OS
11. To understand the different memory management techniques
12. To understand process concurrency and synchronization
13. To understand the concepts of input/ output, storage and file management
14. To study different OS and compare their features.

List of Tutorials

Tutorial No. 1: Operating system services and system calls

Tutorial No. 2: Numerical on process scheduling (non-preemptive)

Tutorial No. 3: Numerical on process scheduling (preemptive)

Tutorial No. 4: Classical problems of synchronization

Tutorial No. 5: Problems on deadlock prevention and Problems on deadlock avoidance

Tutorial No. 6: Virtual to Physical memory translation

Tutorial No. 7: Numerical on memory allocation strategies

Tutorial No. 8: Numerical on page replacement algorithms

Tutorial No. 9: File system structure and its implementation

Tutorial No. 10: Problems on disk scheduling

Text Books:

1. Operating System Principles, Abraham Silberchatz, Peter B. Galvin, Greg Gagne 9th Edition, Wiley Student Edition.
2. Operating systems - Internals and Design Principles, W. Stallings, 6th Edition, Pearson.

Reference Books:

9. Modern Operating Systems, Andrew S Tanenbaum 3rd Edition PHI.
10. Operating Systems A concept - based Approach, 2nd Edition, D. M. Dhamdhere, TMH.
11. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
12. Operating Systems, A. S. Godbole, 2nd Edition, TMH
13. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
14. Operating Systems, S, Haldar and A. A. Arvind, Pearson Education.
15. Operating Systems, R. Elmasri, A. G. Carrick and D. Levine, Mc Graw Hill.
16. Operating Systems in depth, T. W. Doeppner, Wiley.

Course Outcome

7. Compare and contrast different types of operating system and analyze the fundamentals of operating system concepts, services, and components.
8. Analyze and understand operating system process scheduling criteria, scheduling algorithms, threading issues, thread scheduling and inter-process communication mechanism.
9. Analyze process synchronization mechanism by studying and using classical problems of synchronization.
10. Characterize methods and mechanisms to handle, prevent, avoid and detect deadlock in a system and inspect main memory and virtual memory management strategies.
11. Analyze file and directory structure management along with I/O management issues.
12. Carry out case studies in different contemporary operating systems.

FIFTH SEMESTER IT 2018-19 (PATTERN B-16)

COURSE CODE: CS30309 REF NO: To be filled by CD office

Operating System Laboratory

Credits: 1 **Teaching Scheme: -** Laboratory **2 Hrs/Week**

Prerequisites:

1. Computer Organization and Architecture
2. Computer Programming

Objectives:

1. To understand main components of Linux OS and the file system
2. To familiar with shell scripts and the environment
3. To do hands on practice on process creation and synchronization
4. To do hands on practice on system call and different CPU scheduling programs
5. To works for solution to Deadlock situation
6. To have hands on practice on different page replacement algorithm

Course Details:

1. Introduction to Linux and its file system
2. Basic UNIX Commands.
3. UNIX Shell Programming.
4. Programs on process creation and synchronization including shared memory, pipes and messages.(classical problems on synchronization)
5. Programs on UNIX System calls.
6. Programs on CPU Scheduling Algorithms.
7. Programs on Banker's Algorithm for Deadlock Avoidance, Prevention
8. Programs on page replacement algorithm.

List of Practicals:

Experiment No. 1: Introduction to Linux and its File system, File access, mounting.

Experiment No. 2: Basic utility commands of LINUX, File & Directory related commands, basic commands on Linux Administration.

Experiment No. 3: Introduction to shell programming, Control statements

Experiment No. 4: CPU Scheduling Algorithms. (FCFS, RR, SJF, Priority)

Experiment No. 5: Program for FIFO, LRU, and OPTIMAL page replacement algorithm

Experiment No. 6: Programs on process creation, IPC using Pipelines, IPC using message**.**

Experiment No. 8: Thread creation using Pthreads

Experiment No. 9: UNIX: Semaphore, Signals**.**

Experiment No. 10: Program on Banker's Algorithm for Deadlock Avoidance.

Text Books:

1. Unix Concept and Applications 4th Edition Sumitabha Das, The McGraw-Hill Companies
2. UNIX: The Complete Reference, Second Edition, Kenneth Rosen and Douglas Host

Reference Books:

Course Outcome

At the end of the course, the student should be able to:

1. Differentiate Linux file system from other OS.
2. Comparison of the performance of various CPU scheduling algorithms.
3. Critically analyzing the performance of the various page replacement algorithms.
4. Create processes and implement IPC.
5. Implementation of deadlock avoidance algorithms.

COURSE CODE: IT30108 **REF NO: To be filled by CD office**

Data Mining

Credits: 03 **Teaching Scheme:** Theory 03 Hrs/Week

Prerequisites:

- Database Engineering
- Computer Programming

Objective:

- To understand difference between data, information and knowledge
- To gain knowledge on the principles and techniques of data mining and knowledge discovery
- To get familiar to different data mining and web mining techniques

Course Details:

Unit I: Data Mining and Pre-processing (8Hrs)

U1.1 Introduction:

Need of Data Mining, Knowledge Discovery in Database (KDD), Architecture of Data Mining System; Data Objects and Attribute Types, Statistical Description of Data, Data Visualization

U1.2. Data Preprocessing:

Introduction to Data mining, Data mining Functionalities, Data preprocessing (data summarization, data cleaning, data integration and transformation, data reduction, data discretization)

U1.3. Self Study

Integration of Data Mining with a Database or Data Warehouse System, Issues in Data Mining

Unit 2: Mining Frequent Patterns, Association and Correlations (8Hrs)

U2.1 Frequent Itemset Mining:

Interesting Item Set Mining, Market Basket Analysis, Generating Association Rules, Apriori Algorithm, A pattern growth approach for mining frequent item set, Mining frequent item-sets using vertical data, Evaluation of Association Patterns, From Association Analysis to Correlation Analysis

U2.2. Self Study

Sequential Pattern Mining Algorithms, Pattern mining in multi-level, multi-dimensional space Data Integration: different types of digital data and their sources, ETL (extract transform and load) Tools

Unit3: Classification and Prediction (8Hrs)

U3.1 Classification:

Decision Tree Classifier, Rule Based Classification, Bayesian Classification, Neural Network Classification: Back Propagation Algorithm, Lazy Learner: KNN Classifier, Support Vector Machine Classifier Accuracy Measures, Techniques for Evaluating Classifier Accuracy, Ensemble Methods, Multiclass Problem.

U3.2 Prediction:

Linear, Non-Linear Regression.

U3.3 Self Study:

Case-Based Reasoning, Associative Classification, Other Classification Techniques: Genetic Algorithm, Fuzzy Set Approach, Rough Set, Constraints Based Association Mining

Unit 4: Clustering and Outlier Detection (6 Hrs)

U4.1: Cluster Analysis:

Categories of Clustering methods, Different Types of Clusters, Partitioning methods: k-Means, k-Medoids; Hierarchical Clustering Methods: BIRCH, Chameleon; Grid Based Methods: STING; Density based Clustering: DBScan, Cluster Evaluation

U4.2: Outlier Analysis:
Types of outlier, Proximity based approach: distance based, Density based approach
U4.3 Self Study:
Grid Based Methods: CLIQUE, Density based Clustering: OPTICS, Deviation based outlier detection approach: grid based

Unit 5: Advanced Topics in Data Mining (6 Hrs)
U5.1 Web Mining: Introduction, Web Mining, Web Content Mining, Web Structure Mining, Web Usage Mining, Text Mining, Unstructured Text, Episode Rule Discovery for Texts, Hierarchy of Categories, Text Clustering.
U5.2 Temporal and Spatial Data Mining: Introduction, What is Temporal Data Mining? , Temporal Association Rules, Sequence Mining, The GPS Algorithm, SPADE, SPRITE, WUM, Episode Discovery, Event Prediction Problem, Time-Series Analysis, Spatial Mining, Spatial Mining Tasks, Spatial Clustering, Spatial Trends, Conclusion.
U5.3 Self Study:
Graph Mining, Mining Time – Series Data, Multi-relational Data Mining, Data Mining Applications

Text Books

1. "Data Mining: Concepts and Techniques", Jiawei Han and Micheline Kamber, Morgan Kaufman, ISBN 978-81-312-0535-8, 2nd Edition.
2. "Data Mining Techniques", Arun K Pujari, 4st Edition,University Press, 2016.

Reference Books

1. "Data Mining and Analysis: Fundamental Concepts and Algorithms", Mohammed J. Zaki, Wagner Meira Jr., Cambridge University Press.
2. "Mastering Data Mining: The art and science of customer relationship management", M Berry and G. Linoff, John Wiley, ISBN 9971-51-369-2, 2001 Edition.
3. "Data Mining : Theory and Practice" , Soman K P, Diwakar Shyam, Ajay V, New Delhi, Prentice Hall Of India, ISBN 81-203-2897-3, 2006 Edition.
4. "Introduction to Data Mining", Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson 2014.
5. "Data Mining Introductory and advanced topics" , Margaret H Dunham, 6th Edition, Pearson Education, 2009.

Course Outcomes:

Upon completion of the course the graduate students will be able to

1. Identify data mining architecture and different pre-processing techniques required for analysis of given dataset
2. Analyze frequent patterns, determine associations and correlations
3. Apply different classification and prediction to data mining applications
4. Use different clustering mechanisms for data mining
5. Apply data mining for textual, temporal and unstructured data on the Web

COURSE CODE: IT30308 **REF NO: To be filled by CD office**

Data Mining Lab

Credits: 01 **Teaching Scheme:** Lab 02 Hrs/Week

Prerequisites:

- Database Engineering
- Computer Programming

Objective:

- To understand difference between data, information and knowledge
- To gain knowledge on the principles and techniques of data mining and knowledge discovery
- To get familiar to different data mining and web mining techniques

Course Details:

- Basic programming using data mining software such as R (Rattle), WEKA, Rapid Miner, & KNIME (Any One)
- Program for statistical processing of data
- Programs on data pre processing, cleaning and transformations
- Programs on
    - Pattern mining
    - Prediction
    - Cluster analysis
- Report writing using open source documentation software such as Latex
    - Creation of text with header, footer, formatting
    - Creation of Tables
    - Creation of Graphs
    - Mathematical Equation

Course Outcomes:

Upon completion of the course the graduate students will be able to

1. Gain expertise on the use of data mining software
2. Develop programs for Data pre-processing
3. Analyze data mining requirements and develop programs using suitable data mining technique
4. Create reports using suitable software

COURSE CODE: IT34352 **REF NO: To be filled by CD office**

J2EE Enterprise Java Lab

Credits: 01 **Teaching Scheme: -** Laboratory 02 Hrs/Week

Prerequisites:

1) Object Oriented Programming
2) Basic knowledge of Java Programming
3) Web Technology

Objectives:

1. Understand multi-tiered enterprise applications.
2. Understand J2EE framework for developing enterprise applications.
3. Understand various components of J2EE like JSP, Servlets, and effectively use them.
4. Understand Application Server and its configurations.
5. Learn and deploy web based applications in application server.

Course Details:

1. Remote Method Invocation
2. Servlets
3. Java Server Pages
4. Enterprise Java Beans
5. Naming Services, Java Mail and Java Messaging Services
6. Introduction to Struts Framework
7. Introduction to hibernate and HQL
8. Introduction to Spring Framework,
9. Web services

List of Experiments:

Experiment No. 1: Create a RMI Program showing marshalling and un-marshalling processes.

Experiment No. 2: Log in application using servlet and JDBC.

Experiment No. 3: Implement session tracking techniques in servlets.

Experiment No. 4: Log in application using JSP and JDBC.

.

Experiment No. 6: Implement session tracking techniques in JSP.

Experiment No. 7: Implementation of session, message-driven and entity bean using EJB.

Experiment No. 8: Programs on

i. Implementation of JMS to send mail

ii. Implementation of Java Message Service to send SMS.

Experiment No. 9: Programs on
- **i.** Creating registration form using Struts
- **ii.** Implementing Dependency injection and inversion of control

Experiment No. 10: Develop a web application using hibernate to maintain student data.

Experiment No. 11: Create a web application for ticket booking using spring.

Experiment No. 12: Creation of web services with JAX-WS.

Text Books:

1. Kongent S., "Java Server Programming (JEE 6) Black Book, Platinum Edition", 2008, Dreamtech / Wiley India Pvt. Ltd.
2. Eric Jendrock, D. Carson, I. Evans, D. Gollapudi, K. Haase, C. Srivastha, "The Java EE6 Tutorial", Volume-1, Fourth Edition, 2010, Pearson India, New Delhi.

Reference Books:

1. Douglas E. Comer, "Internetworking with TCP/IP, Volume 1: Principles, Protocols and Architecture", Fifth Edition, 2006, PHI Learning Pvt. Ltd., New Delhi.
2. SANTOSH KUMAR K , "Jdbc, Servlets, And Jsp Black Book" Black Book, New Edition, 2008

Course Outcomes:

Upon completion of the course, graduates will be able to-

1. Use RMI for invoking remote methods for user benefit.
2. Implement different applications using Servlets and JSPs.
3. Use MVC architecture through EJB
4. Develop applications through Struts & Spring frameworks