SCHEME OF TEACHING AND EXAMINATION
B.E. COMPUTER SCIENCE AND ENGINEERING

V SEMESTER

S. No.	Subject Code	Subject	Teaching Dept.	Teaching Hrs / Week		Examination			
				Theory	Practical	Duration (Hrs)	Marks		
							IA	Exam	Total
1	10IS51	Software Engineering	CSE/ISE	04	-	03	25	100	125
2	10CS52	Systems Software	CSE/ISE	04	-	03	25	100	125
3	10CS53	Operating Systems	CSE/ISE	04	-	03	25	100	125
4	10CS54	Database Management Systems	CSE/ISE	04	-	03	25	100	125
5	10CS55	Computer Networks - I	CSE/ISE	04	-	03	25	100	125
6	10CS56	Formal Languages and Automata Theory	CSE/ISE	04	-	03	25	100	125
7	10CSL57	Database Applications Laboratory	CSE/ISE	-	03	03	25	50	75
8	10CSL58	Systems Software & Operating Systems Laboratory	CSE/ISE	-	03	03	25	50	75
		Total		24	06	-	200	700	900

4

stop. If some requests occur during going up or coming down they should be ignored.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

V SEMESTER

SOFTWARE ENGINEERING

Subject Code: 10IS51	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT – 1   **6 Hours**
Overview: Introduction: FAQ's about software engineering, Professional and ethical responsibility.
Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.

UNIT – 2   **6 Hours**
Critical Systems, Software Processes: Critical Systems: A simple safety-critical system; System dependability; Availability and reliability.
Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering.

UNIT – 3   **7 Hours**
Requirements: Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document.
Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.

UNIT – 4   **7 Hours**
System models, Project Management: System Models: Context models; Behavioral models; Data models; Object models; Structured methods.
Project Management: Management activities; Project planning; Project scheduling; Risk management.

PART - B

UNIT – 5

Software Design : Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles.
Object-Oriented design: Objects and Object Classes; An Object-Oriented design process; Design evolution.

7 Hours

UNIT – 6

Development: Rapid Software Development: Agile methods; Extreme programming; Rapid application development.
Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.

6 Hours

UNIT – 7

Verification and Validation: Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods.
Software testing: System testing; Component testing; Test case design; Test automation.

7 Hours

UNIT – 8

Management: Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model.
Software Cost Estimation: Productivity; Estimation techniques; Algorithmic cost modeling, Project duration and staffing.

6 Hours

Text Books:

1. Ian Sommerville: Software Engineering, 8th Edition, Pearson Education, 2007.
   (Chapters-: 1, 2, 3, 4, 5, 6, 7, 8, 11, 14, 17, 21, 22, 23, 25, 26)

Reference Books:

1. Roger.S.Pressman: Software Engineering-A Practitioners approach, 7th Edition, McGraw Hill, 2007.
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India, 2009.

SYSTEM SOFTWARE

Subject Code: 10CS52   I.A. Marks : 25
Hours/Week : 04   Exam Hours: 03
Total Hours : 52   Exam Marks: 100

PART – A

UNIT – 1

6 Hours

34

Machine Architecture: Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.

UNIT – 2   6 Hours
Assemblers -1: Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation.

UNIT – 3   6 Hours
Assemblers -2: Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples - MASM Assembler.

UNIT – 4   8 Hours
Loaders and Linkers: Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples - MS-DOS Linker.

PART – B

UNIT – 5   6 Hours
Editors and Debugging Systems: Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria

UNIT – 6   8 Hours
Macro Processor: Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options - Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.

UNIT – 7   6 Hours

Lex and Yacc – 1: Lex and Yacc - The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, The Parts of Speech Lexer, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.

UNIT – 8   6 Hours
Lex and Yacc - 2
Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Tokens.

Text Books:
1. Leland.L.Beck: System   Software,   3rd Edition,   Pearson Education, 1997.
   (Chapters 1.1 to 1.3, 2 (except 2.5.2 and 2.5.3), 3 (except 3.5.2 and 3.5.3), 4 (except 4.4.3))
2. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 1998.
   (Chapters 1, 2 (Page 2-42), 3 (Page 51-65))

Reference Books:
1. D.M.Dhamdhere: System Programming and Operating Systems, 2nd Edition, Tata McGraw - Hill, 1999.

OPERATING SYSTEMS

Subject Code: 10CS53	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT – 1   6 Hours
Introduction to Operating Systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments.Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System

36

design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

UNIT – 2   **7 Hours**
Process Management: Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.

UNIT – 3   **7 Hours**
Process Synchronization : Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

UNIT – 4   **6 Hours**
Deadlocks: Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

PART – B

UNIT – 5   **7 Hours**
Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

UNIT – 6   **7 Hours**
File System, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

UNIT – 7   **6 Hours**
Secondary Storage Structures, Protection : Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

UNIT – 8   **6 Hours**
Case Study: The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication.

Text Books:
1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
   (Chapters: 1, 2, 3.1 to 3.4 , 4.1 to 4.4, 5.1 to 5.5, 6.1 to 6.7, 7, 8.1 to 8.6, 9.1 to 9.6, 10, 11.1 to 11.5, 12.1 to 12.6, 17.1 to 17.8, 21.1 to 21.9)

Reference Books:
1. D.M Dhamdhere: Operating systems - A concept based Approach, 2nd Edition, Tata McGraw- Hill, 2002.
2. P.C.P. Bhatt: Introduction to Operating Systems: Concepts and Practice, 2nd Edition, PHI, 2008.
3. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 1990.

38

DATABASE MANAGEMENT SYSTEMS

Subject Code: 10CS54   I.A. Marks : 25
Hours/Week : 04   Exam Hours: 03
Total Hours : 52   Exam Marks: 100

PART - A

UNIT – 1   6 Hours

Introduction: Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS.

Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems.

UNIT – 2   6 Hours

Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.

UNIT – 3   8 Hours

Relational Model and Relational Algebra : Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.

UNIT – 4   6 Hours

SQL – 1: SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL Queries.

PART - B

UNIT – 5   **6 Hours**

SQL – 2: Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL / PSM.

UNIT – 6   **6 Hours**

Database Design – 1: Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form

UNIT – 7   **6 Hours**

Database Design -2: Properties of Relational Decompositions; Algorithms for Relational Database Schema Design; Multivalued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form; Inclusion Dependencies; Other Dependencies and Normal Forms

UNIT – 8   **8 Hours**

Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Introduction to ARIES; The log; Other recovery-related structures; The write-ahead log protocol; Checkpointing; Recovering from a System Crash; Media Recovery; Other approaches and interaction with concurrency control.

Text Books:

1. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Pearson Education, 2007.
    (Chapters 1, 2, 3 except 3.8, 5, 6.1 to 6.5, 7.1, 8, 9.1, 9.2 except SQLJ, 9.4, 10)
2. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.
    (Chapters 16, 17.1, 17.2, 18)

Reference Books:

1. Silberschatz, Korth and Sudharshan: Data base System Concepts, 6th Edition, Mc-GrawHill, 2010.
2. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson Education, 2006.

40

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

COMPUTER NETWORKS - I

Subject Code: 10CS55	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT - 1   7 Hours

Introduction: Data Communications, Networks, The Internet, Protocols & Standards, Layered Tasks,
The OSI model, Layers in OSI model, TCP/IP Protocol suite, Addressing

UNIT- 2   7 Hours

Physical Layer-1: Analog & Digital Signals, Transmission Impairment, Data Rate limits, Performance, Digital-digital conversion (Only Line coding: Polar, Bipolar and Manchester coding), Analog-to-digital conversion (only PCM), Transmission Modes, Digital-to-analog conversion

UNIT- 3   6 Hours

Physical Layer-2 and Switching: Multiplexing, Spread Spectrum, Introduction to switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks

UNIT- 4   6 Hours

Data Link Layer-1: Error Detection & Correction: Introduction, Block coding, Linear block codes, Cyclic codes, Checksum.

PART - B

UNIT- 5   6 Hours

Data Link Layer-2: Framing, Flow and Error Control, Protocols, Noiseless Channels, Noisy channels, HDLC, PPP (Framing, Transition phases only)

UNIT- 6   7 Hours

Multiple Access & Ethernet: Random access, Controlled Access, Channelization, Ethernet: IEEE standards, Standard
Ethernet, Changes in the standard, Fast Ethernet, Gigabit Ethernet

UNIT - 7   6 Hours

Wireless LANs and Cellular Networks: Introduction, IEEE 802.11, Bluetooth, Connecting devices, Cellular Telephony

UNIT - 8:   **7 Hours**

Network Layer: Introduction, Logical addressing, IPv4 addresses, IPv6 addresses, Internetworking basics,
IPv4, IPv6, Comparison of IPv4 and IPv6 Headers.

Text Books:
1. Behrouz A. Forouzan,: Data Communication and Networking, 4th Edition Tata McGraw-Hill, 2006.
   (Chapters 1.1 to 1.4, 2.1 to 2.5, 3.1 To 3.6, 4.1 to 4.3, 5.1, 6.1, 6.2,   8.1 to 8.3, 10.1 to 10.5, 11.1 to 11.7, 12.1 to 12.3, 13.1 to 13.5,   14.1, 14.2, 15.1, 16.1, 19.1, 19.2, 20.1 to 20.3)

Reference Books:
1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks - Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
4. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007.

FORMAL LANGUAGES AND AUTOMATA THEORY

Subject Code: 10CS56   I.A. Marks : 25
Hours/Week : 04   Exam Hours: 03
Total Hours : 52   Exam Marks: 100

PART - A

UNIT – 1   **7 Hours**
Introduction to Finite Automata: Introduction to Finite Automata; The central concepts of Automata theory; Deterministic finite automata; Nondeterministic finite automata

UNIT – 2   **7 Hours**
Finite Automata, Regular Expressions: An application of finite automata; Finite automata with Epsilon-transitions; Regular expressions; Finite Automata and Regular Expressions; Applications of Regular Expressions

42

UNIT – 3   **6 Hours**

Regular Languages, Properties of Regular Languages: Regular languages; Proving languages not to be regular languages; Closure properties of regular languages; Decision properties of regular languages; Equivalence and minimization of automata

UNIT – 4   **6 Hours**

Context-Free Grammars And Languages : Context –free grammars; Parse trees; Applications; Ambiguity in grammars and Languages .

PART – B

UNIT – 5   **7 Hours**

Pushdown Automata: Definition of the Pushdown automata; the languages of a PDA; Equivalence of ˜ PDA's and CFG's; Deterministic Pushdown Automata

UNIT – 6   **6 Hours**

Properties of Context-Free Languages: Normal forms for CFGs; The pumping lemma for CFGs; Closure properties of CFLs

UNIT – 7   **7 Hours**

Introduction To Turing Machine: Problems that Computers cannot solve; The turning machine; Programming techniques for Turning Machines; Extensions to the basic Turning Machines; Turing Machine and Computers.

UNIT – 8   **6 Hours**

Undecidability: A Language that is not recursively enumerable; An Undecidable problem that is RE; Post's Correspondence problem; Other undecidable problems.

Text Books:
1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson Education, 2007.
   (Chapters: 1.1, 1.5, 2.2 to 2.5, 3.1 to 3.3, 4, 5, 6, 7, 8.1 to 8.4, 8.6, 9.1, 9.2, 9.4.1, 9.5)

Reference Books:
1. K.L.P. Mishra: Theory of Computer Science, Automata, Languages, and Computation, 3rd Edition, PHI, 2007.
2. Raymond Greenlaw, H.James Hoover: Fundamentals of the Theory of Computation, Principles and Practice, Morgan Kaufmann, 1998.

43

Rahul

H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

3. John C Martin: Introduction to Languages and Automata Theory, 3rd Edition, Tata McGraw-Hill, 2007.
4. Thomas A. Sudkamp: An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition, Pearson Education, 2006.

DATABASE APPLICATIONS LABORATORY

Subject Code: 10CSL57   I.A. Marks   : 25
Hours/Week : 03   Exam Hours: 03
Total Hours : 42   Exam Marks: 50

1. Consider the following relations:
    Student (*snum:* integer, *sname:* string, *major:* string, *level:* string, *age:* integer)
    Class (*name:* string, *meets at:* string, *room:* string, *d:* integer)
    Enrolled (*snum:* integer, *cname:* string)
    Faculty (fid: integer, *fname:* string, *deptid:* integer)
    The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)
    Write the following queries in SQL. No duplicates should be printed in any of the answers.

    i.   Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. Harshith
    ii.   Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
    iii. Find the names of all students who are enrolled in two classes that meet at the same time.
    iv.   Find the names of faculty members who teach in every room in which some class is taught.
    v.   Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

2. The following relations keep track of airline flight information:
    Flights (*no:* integer, *from:* string, *to:* string, *distance:* integer, *Departs:* time, *arrives:* time, *price:* real)
    Aircraft (*aid:* integer, *aname:* string, *cruisingrange:* integer)
    Certified (*eid:* integer, *aid:* integer)
    Employees (*eid:* integer, *ename:* string, *salary:* integer)
    Note that the Employees relation describes pilots and other kinds of employees as well;   Every pilot is certified for some aircraft, and only pilots are certified to fly.

44

Write each of the following queries in SQL.

    i.   Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80, 000.

    ii.   For each pilot who is certified for more than three aircrafts, find the *eid* and the maximum *cruisingrange* of the aircraft for which she or he is certified.

    iii.   Find the names of pilots whose *salary* is less than the price of the cheapest route from Bengaluru to Frankfurt.

    iv.   For all aircraft with *cruisingrange* over 1000 Kms, .find the name of the aircraft and the average salary of all pilots certified for this aircraft.

    v.   Find the names of pilots certified for some Boeing aircraft.

    vi.   Find the *aids* of all aircraft that can be used on routes from Bengaluru to New Delhi.

3. Consider the following database of student enrollment in courses & books adopted for each course.
STUDENT (regno: string, name: string, major: string, bdate:date)
COURSE (course #:int, cname:string, dept:string)
ENROLL (regno:string, course#:int, sem:int, marks:int)
BOOK _ ADOPTION (course# :int, sem:int, book-ISBN:int)
TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)

    i.   Create the above tables by properly specifying the primary keys and the foreign keys.

    ii.   Enter at least five tuples for each relation.

    iii.   Demonstrate how you add a new text book to the database and make this book be adopted by some department.

    iv.   Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.

    v.   List any department that has *all* its adopted books published by a specific publisher.

    vi.   Generate suitable reports.

    vii.   Create suitable front end for querying and displaying the results.

4. The following tables are maintained by a book dealer.
AUTHOR (author-id:int, name:string, city:string, country:string)
PUBLISHER (publisher-id:int, name:string, city:string, country:string)
CATALOG (book-id:int, title:string, author-id:int, publisher-id:int, category-id:int, year:int, price:int)
CATEGORY (category-id:int, description:string)

45

ORDER-DETAILS (<u>order-no</u>:int, <u>book-id</u>:int, quantity:int)

i. Create the above tables by properly specifying the primary keys and the foreign keys.
ii. Enter at least five tuples for each relation.
iii. Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
iv. Find the author of the book which has maximum sales.
v. Demonstrate how you increase the price of books published by a specific publisher by 10%.
vi. Generate suitable reports.
vii. Create suitable front end for querying and displaying the results.

5. Consider the following database for a banking enterprise
BRANCH(<u>branch-name</u>:string, branch-city:string, assets:real)
ACCOUNT(<u>accno</u>:int, branch-name:string, balance:real)
DEPOSITOR(<u>customer-name</u>:string, <u>accno</u>:int)
CUSTOMER(<u>customer-name</u>:string, customer-street:string, customer-city:string)
LOAN(<u>loan-number</u>:int, branch-name:string, amount:real)
BORROWER(<u>customer-name</u>:string, <u>loan-number</u>:int)

i. Create the above tables by properly specifying the primary keys and the foreign keys
ii. Enter at least five tuples for each relation
iii. Find all the customers who have at least two accounts at the *Main* branch.
iv. Find all the customers who have an account at *all* the branches located in a specific city.
v. Demonstrate how you delete all account tuples at every branch located in a specific city.
vi. Generate suitable reports.
vii. Create suitable front end for querying and displaying the results.

Instructions:
1. The exercises are to be solved in an RDBMS environment like Oracle or DB2.
2. Suitable tuples have to be entered so that queries are executed correctly.
3. Front end may be created using either VB or VAJ or any other similar tool.
4. The student need not create the front end in the examination. The results of the queries may be displayed directly.

5. Relevant queries other than the ones listed along with the exercises may also be asked in the examination.
6. Questions must be asked based on lots.

SYSTEM SOFTWARE & OPERATING SYSTEMS LABORATORY

Subject Code: 10CSL58   I.A. Marks : 25
Hours/Week : 03   Exam Hours: 03
Total Hours : 42   Exam Marks: 50

PART - A

LEX and YACC Programs:
Design, develop, and execute the following programs using LEX:

1. a) Program to count the number of characters, words, spaces and lines in a given input file.
   b) Program to count the numbers of comment lines in a given C program. Also   eliminate them and copy the resulting program into separate file.

2. a) Program to recognize a valid arithmetic expression and to recognize the identifiers and operators present. Print them separately.
   b) Program to recognize whether a given sentence is simple or compound.

3. Program to recognize and count the number of identifiers in a given input file.

Design, develop, and execute the following programs using YACC:

4. a) Program to recognize a valid arithmetic expression that uses operators +, -, * and /.
   b) Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.

5. a) Program to evaluate an arithmetic expression involving operators +, -, * and /.
   b) Program to recognize strings 'aaab', 'abbb', 'ab' and 'a' using the grammar $(a^n b^n, n >= 0)$.

6. Program to recognize the grammar $(a^n b, n >= 10)$.

Signature

H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

PART B

UNIX Programming:

Design, develop, and execute the following programs:

7. a) Non-recursive shell script that accepts any number of arguments and prints them in the Reverse order, (For example, if the script is named rargs, then executing rargs A B C should produce C B A on the standard output).

    b) C program that creates a child process to read commands from the standard input and execute them (a minimal implementation of a shell – like program). You can assume that no arguments will be passed to the commands to be executed.

8. a) Shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, outputs the common permissions, otherwise outputs each file name followed by its permissions.

    b) C program to create a file with 16 bytes of arbitrary data from the beginning and another 16 bytes of arbitrary data from an offset of 48. Display the file contents to demonstrate how the hole in file is handled.

9. a) Shell script that accepts file names specified as arguments and creates a shell script that contains this file as well as the code to recreate these files. Thus if the script generated by your script is executed, it would recreate the original files(This is same as the "bundle" script described by Brain W. Kernighan and Rob Pike in " The Unix Programming Environment", Prentice – Hall India).

    b) C program to do the following: Using fork( ) create a child process. The child process prints its own process-id and id of its parent and then exits. The parent process waits for its child to finish (by executing the wait( )) and prints its own process-id and the id of its child process and then exits.

Operating Systems:

10. Design, develop and execute a program in C / C++ to simulate the working of Shortest Remaining Time and Round-Robin Scheduling Algorithms. Experiment with different quantum sizes for the Round-Robin algorithm. In all cases, determine the average turn-around time. The input can be read from key board or from a file.

11. Using OpenMP, Design, develop and run a multi-threaded program to generate and print Fibonacci Series. One thread has to generate

SCHEME OF TEACHING AND EXAMINATION
B.E. COMPUTER SCIENCE AND ENGINEERING

VI SEMESTER

S. No.	Subject Code	Subject	Teaching Dept.	Teaching Hrs / Week		Examination			
				Theory	Practical	Duration (Hrs)	IA	Exam	Total
1	10AL61	Management and Entrepreneurship	CSE/ISE/ MBA	04	-	03	25	100	125
2	10CS62	Unix System Programming	CSE/ISE	04	-	03	25	100	125
3	10CS63/ 10IS662	Compiler Design	CSE/ISE	04	-	03	25	100	125
4	10CS64	Computer Networks - II	CSE/ISE	04	-	03	25	100	125
5	10CS65 / 10IS665	Computer Graphics and Visualization	CSE/ISE	04	-	03	25	100	125
6	10CS66x	Elective I (Group-A)	CSE/ISE	04	-	03	25	100	125
7	10CSL67	Computer Graphics and Visualization Laboratory	CSE/ISE	-	03	03	25	50	75
8	10CSL68	Unix System Programming and Compiler Design Laboratory	CSE/ISE	-	03	03	25	50	75
		Total		24	06	-	200	700	900

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

5

10CS661/10IS661	Operations Research
10CS662	Signals and Systems
10CS663/10IS663	Data Compression
10CS664/10IS664	Pattern Recognition
10CS665	Stochastic Models and Applications
10CS666/10IS666	Programming Languages

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

6

Syllabus

VI SEMESTER

MANAGEMENT & ENTREPRENEURSHIP

Subject Code : **10AL61**   IA Marks : 25
No. of Lecture Hrs/Week : 04   Exam Hours : 03
Total no. of Lecture Hrs. : 52   Exam Marks : 100

MANAGEMENT (PART – A)

UNIT - 1

MANAGEMENT: Introduction - Meaning - nature and characteristics of Management, Scope and functional areas of Management - Management as a Science, Art or Profession Management & Administration - Roles of Management, Levels of Management, Development of Management Thought-Early Management Approaches-Modern Management Approaches.
                                                                                   7 Hours

UNIT - 2

PLANNING: Nature, importance and purpose of planning process Objectives - Types of plans (Meaning only) - Decision making – Importance of planning - steps in planning & planning premises - Hierarchy of plans.   **6 Hours**

UNIT - 3

ORGANISING AND STAFFING: Nature and purpose of organization Principles of organization - Types of organization - Departmentation Committees – Centralisation Vs Decentralisation of authority and responsibility - Span of control - MBO and MBE (Meaning only) Nature and importance of Staffing - Process of Selection & Recruitment (in brief).
                                                                                   7 Hours

UNIT - 4

DIRECTING & CONTROLLING: Meaning and nature of directing Leadership styles, Motivation Theories, Communication - Meaning and importance – Coordination, meaning and importance and Techniques of Co -ordination. Meaning and steps in controlling - Essentials of a sound control system - Methods of establishing control.   **6 Hours**

ENTREPRENEURSHIP (PART – B)

UNIT - 5

ENTREPRENEUR: Meaning of Entrepreneur; Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneur, Intrapreneur – an emerging Class. Concept of Entrepreneurship - Evolution of Entrepreneurship, Development of Entrepreneurship; Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship – its Barriers.   **6 Hours**

UNIT - 6

SMALL SCALE INDUSTRY: Definition; Characteristics; Need and rationale: Objectives; Scope; role of SSI in Economic Development. Advantages of SSI Steps to start an SSI - Government policy towards SSI; Different Policies of S.S.I.; Government Support for S.S.I. during 5 year plans, Impact of Liberalization, Privatization, Globalization on S.S.I., Effect of WTO/GATT Supporting Agencies of Government for S.S.I Meaning; Nature of Support; Objectives; Functions; Types of Help; Ancillary Industry and Tiny Industry (Definition only).
                                                                                   7 Hours

UNIT - 7

INSTITUTIONAL SUPPORT: Different Schemes; TECKSOK; KIADB; KSSIDC; KSIMC; DIC Single Window Agency: SISI; NSIC; SIDBI; KSFC.   **6 Hours**

UNIT - 8

PREPARATION OF PROJECT: Meaning of Project; Project Identification; Project Selection; Project Report; Need and Significance of Report; Contents; formulation; Guidelines by Planning Commission for Project report; Network Analysis; Errors of Project Report; Project Appraisal. Identification of Business Opportunities - Market Feasibility Study; Technical Feasibility Study; Financial Feasibility Study & Social Feasibility Study.   **7 Hours**

TEXT BOOKS:

1. **Principles of Management** - P. C. Tripathi, P. N. Reddy; Tata McGraw Hill, 4th Edition, 2010.
2. **Dynamics of Entrepreneurial Development & Management** – Vasant Desai Himalaya Publishing House.
3. **Entrepreneurship Development** - Small Business Enterprises Poornima M Charantimath - Pearson Education – 2006.

REFERENCE BOOKS:

1. **Management Fundamentals** - Concepts, Application, Skill Development Robert Lusier – Thomson.
2. **Entrepreneurship Development** - S S Khanka - S Chand & Co.
3. **Management** - Stephen Robbins - Pearson Education /PHI -17th Edition, 2003.

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

the numbers up to the specified limit and another thread has to print them. Ensure proper synchronization.

12. Design, develop and run a program to implement the Banker's Algorithm. Demonstrate its working with different data values.

Instructions:

In the examination, a combination of one LEX and one YACC problem has to be asked from Part A for a total of 30 marks and one programming exercise from Part B has to be asked for a total of 20 marks.

VI SEMESTER

MANAGEMENT AND ENTREPRENEURSHIP
(Common to All Branches)

Subject Code: 10AL61	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

UNIX SYSTEM PROGRAMMING

Subject Code: 10CS62	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART - A

UNIT – 1   **6 Hours**
Introduction: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards.
UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.

UNIT – 2   **6 Hours**
UNIX Files: File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.

49

UNIT – 3   **7 Hours**

UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs, General File Class, regfile Class for Regular Files, dirfile Class for Directory Files, FIFO File Class, Device File Class, Symbolic Link File Class, File Listing Program.

UNIT – 4   **7 Hours**

UNIX Processes: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.

PART - B

UNIT – 5   **7 Hours**

Process Control : Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.

Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.

UNIT – 6   **7 Hours**

Signals and Daemon Processes: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers.

Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.

UNIT – 7   **6 Hours**

Interprocess Communication – 1: Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores.

UNIT – 8   **6 Hours**

Interprocess Communication – 2: Shared Memory, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.

50

Text Books:

1. Terrence Chan: UNIX System Programming Using C++, Prentice Hall India, 1999.
   (Chapters 1, 5, 6, 7, 8, 9, 10)
2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005.
   (Chapters 7, 8, 9, 13, 14, 15)

Reference Books:

1. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson Education, 2005.
2. Maurice J Bach: The Design of the UNIX Operating System, Pearson Education, 1987.
3. Uresh Vahalia: UNIX Internals: The New Frontiers, Pearson Education, 2001.

FILE STRUCTURES

Subject Code: 10IS63	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT – 1   7 Hours

Introduction: File Structures: The Heart of the file structure Design, A Short History of File Structure Design, A Conceptual Toolkit; Fundamental File Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters, The Unix Directory Structure, Physical devices and Logical Files, File-related Header Files, UNIX file System Commands; Secondary Storage and System Software: Disks, Magnetic Tape, Disk versus Tape; CD-ROM: Introduction, Physical Organization, Strengths and Weaknesses; Storage as Hierarchy, A journey of a Byte, Buffer Management, Input /Output in UNIX.

UNIT – 2   6 Hours

Fundamental File Structure Concepts, Managing Files of Records :
Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length, Fixed Field Buffers, An Object-Oriented Class for Record Files, Record Access, More about Record Structures, Encapsulating Record Operations in a Single Class, File Access and File Organization.

51

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

UNIT – 3
7 Hours

Organization of Files for Performance, Indexing: Data Compression, Reclaiming Space in files, Internal Sorting and Binary Searching, Keysorting; What is an Index? A Simple Index for Entry-Sequenced File, Using Template Classes in C++ for Object I/O, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index structure: Inverted Lists, Selective indexes, Binding.

UNIT – 4
6 Hours

Cosequential Processing and the Sorting of Large Files: A Model for Implementing Cosequential Processes, Application of the Model to a General Ledger Program, Extension of the Model to include Mutiway Merging, A Second Look at Sorting in Memory, Merging as a Way of Sorting Large Files on Disk.

PART - B

UNIT – 5
7 Hours

Multi-Level Indexing and B-Trees: The invention of B-Tree, Statement of the problem, Indexing with Binary Search Trees; Multi-Level Indexing, B-Trees, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods; Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution, Redistribution during insertion; B* Trees, Buffering of pages; Virtual B-Trees; Variable-length Records and keys.

UNIT – 6
6 Hours

Indexed Sequential File Access and Prefix B + Trees: Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Listead of Keys, The Simple Prefix B+ Tree and its maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B- Tree, Loading a Simple Prefix B+ Trees, B-Trees, B+ Trees and Simple Prefix B+ Trees in Perspective.

UNIT – 7
7 Hours

Hashing: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, How much Extra Memory should be used?, Collision resolution by progressive overflow, Buckets, Making deletions, Other collision resolution techniques, Patterns of record access.

UNIT – 8
6 Hours

Extendible Hashing: How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches.

Text Books:
1. Michael J. Folk, Bill Zoellick, Greg Riccardi: File Structures-An Object Oriented Approach with C++, 3rd Edition, Pearson Education, 1998.
(Chapters 1 to 12 excluding 1.4, 1.5, 5.5, 5.6, 8.6, 8.7, 8.8)

Reference Books:
1. K.R. Venugopal, K.G. Srinivas, P.M. Krishnaraj: File Structures Using C++, Tata McGraw-Hill, 2008.
2. Scot Robert Ladd: C++ Components and Algorithms, BPB Publications, 1993.
3. Raghu Ramakrishan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw Hill, 2003.

COMPUTER NETWORKS - II

Subject Code: 10CS64	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART - A

UNIT - 1   **6 Hours**
Packet Switching Networks - 1: Network services and internal network operation, Packet network topology, Routing in Packet networks, Shortest path routing: Bellman-Ford algorithm.

UNIT – 2   **6 Hours**
Packet Switching Networks – 2: Shortest path routing (continued), Traffic management at the Packet level, Traffic management at Flow level, Traffic management at flow aggregate level.

UNIT – 3   **6 Hours**
TCP/IP-1: TCP/IP architecture, The Internet Protocol, IPv6, UDP.

UNIT – 4   **8 Hours**
TCP/IP-2: TCP, Internet Routing Protocols, Multicast Routing, DHCP, NAT and Mobile IP.

53

PART – B

UNIT - 5   **7 Hours**

Applications, Network Management, Network Security: Application layer overview, Domain Name System (DNS), Remote Login Protocols, E-mail, File Transfer and FTP, World Wide Web and HTTP, Network management, Overview of network security, Overview of security methods, Secret-key encryption protocols, Public-key encryption protocols, Authentication, Authentication and digital signature, Firewalls.

UNIT – 6   **6 Hours**

QoS, VPNs, Tunneling, Overlay Networks: Overview of QoS, Integrated Services QoS, Differentiated services QoS, Virtual Private Networks, MPLS, Overlay networks.

UNIT - 7   **7 Hours**

Multimedia Networking: Overview of data compression, Digital voice and compression, JPEG, MPEG, Limits of compression with loss, Compression methods without loss, Overview of IP Telephony, VoIP signaling protocols, Real-Time Media Transport Protocols, Stream control Transmission Protocol (SCTP)

UNIT – 8   **6 Hours**

Mobile AdHoc Networks and Wireless Sensor Neworks: Overview of Wireless Ad-Hoc networks, Routing in AdHOc Networks, Routing protocols for and Security of AdHoc networks, Sensor Networks and protocol structures, Communication Energy model, Clustering protocols, Routing protocols, ZigBee technology and 802.15.4.

Text Books:

1. Communication Networks – Fundamental Concepts & key architectures, Alberto Leon Garcia & Indra Widjaja, 2nd Edition, Tata McGraw-Hill, India
   (7 - excluding 7.6, 8)
2. Computer & Communication Networks, Nadir F Mir, Pearson Education, India
   (9, 10 excluding 10.7, 12.1 to 12.3, 16, 17.1 to 17.6, 18.1 to18.3, 18.5, 19, 20)

54

Reference Books:
1. Behrouz A. Forouzan: Data Communications and Networking, 4th Edition, Tata McGraw-Hill, 2006.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L Peterson and Bruce S Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
4. Wayne Tomasi: Introduction to Data Communications and Networking, Pearson Education, 2005.

SOFTWARE TESTING

Subject Code: 10IS65	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT 1 6 Hours
A Perspective on Testing, Examples: Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudocode, The triangle problem, The NextDate function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper.

UNIT 2 7 Hours
Boundary Value Testing, Equivalence Class Testing, Decision Table-Based Testing: Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations.

UNIT 3 7 Hours
Path Testing, Data Flow Testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations. Definition-Use testing, Slice-based testing, Guidelines and observations.

UNIT 4 6 Hours
Levels of Testing, Integration Testing: Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and

55

Sreehit

H.O.D.
Dept. Of Information Science & Engineering
Aiva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

system testing. A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations

PART – B

UNIT 5   7 Hours
System Testing, Interaction Testing: Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example. Context of interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing,.

UNIT 6   7 Hours
Process Framework: Validation and verification, Degrees of freedom, Varieties of software. Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback. The quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis, Testing, Improving the process, Organizational factors.

UNIT 7   6 Hours
Fault-Based Testing, Test Execution: Overview, Assumptions in fault-based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. Test Execution: Overview, from test case specifications to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay.

UNIT 8   6 Hours
Planning and Monitoring the Process, Documenting Analysis and Test: Quality and process, Test and analysis strategies and plans, Risk planning, Monitoring the process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

TEXT BOOKS:
1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2008.
   (Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 1314, 15)
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2008.
   (Listed topics only from Chapters 2, 3, 4, 16, 17, 20, 24)

56

REFERENCE BOOKS:
1. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.
2. Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 2nd Edition, Pearson Education, 2007.
3. Brian Marrick: The Craft of Software Testing, Pearson Education, 1995.

OPERATIONS RESEARCH

Subject Code: 10IS661	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART - A

UNIT – 1   **6 Hours**
Introduction, Linear Programming – 1: Introduction: The origin, nature and impact of OR; Defining the problem and gathering data; Formulating a mathematical model; Deriving solutions from the model; Testing the model; Preparing to apply the model; Implementation .
Introduction to Linear Programming: Prototype example; The linear programming (LP) model.

UNIT – 2   **7 Hours**
LP – 2, Simplex Method – 1: Assumptions of LP; Additional examples.
The essence of the simplex method; Setting up the simplex method; Algebra of the simplex method; the simplex method in tabular form; Tie breaking in the simplex method

UNIT – 3   **6 Hours**
Simplex Method – 2: Adapting to other model forms; Post optimality analysis; Computer implementation
Foundation of the simplex method.

UNIT – 4   **7 Hours**
Simplex Method – 2, Duality Theory: The revised simplex method, a fundamental insight.
The essence of duality theory; Economic interpretation of duality, Primal dual relationship; Adapting to other primal forms

57

UNIT – 5 7 Hours

Duality Theory and Sensitivity Analysis, Other Algorithms for LP : The role of duality in sensitive analysis; The essence of sensitivity analysis; Applying sensitivity analysis. The dual simplex method; Parametric linear programming; The upper bound technique.

UNIT – 6 7 Hours

Transportation and Assignment Problems: The transportation problem; A streamlined simplex method for the transportation problem; The assignment problem; A special algorithm for the assignment problem.

UNIT – 7 6 Hours

Game Theory, Decision Analysis: Game Theory: The formulation of two persons, zero sum games; Solving simple games- a prototype example; Games with mixed strategies; Graphical solution procedure; Solving by linear programming, Extensions.
Decision Analysis: A prototype example; Decision making without experimentation; Decision making with experimentation; Decision trees.

UNIT – 8 6 Hours

Metaheuristics: The nature of Metaheuristics, Tabu Search, Simulated Annealing, Genetic Algorithms.

Text Books:
1. Frederick S. Hillier and Gerald J. Lieberman: Introduction to Operations Research: Concepts and Cases, 8th Edition, Tata McGraw Hill, 2005.
   (Chapters: 1, 2, 3.1 to 3.4, 4.1 to 4.8, 5, 6.1 to 6.7, 7.1 to 7.3, 8, 13, 14, 15.1 to 15.4)

Reference Books:
1. Wayne L. Winston: Operations Research Applications and Algorithms, 4th Edition, Cengage Learning, 2003.
2. Hamdy A Taha: Operations Research: An Introduction, 8th Edition, Pearson Education, 2007.

58

H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

COMPILER DESIGN

Subject Code: 10IS662
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT – 1   8 Hours
Introduction, Lexical analysis: Language processors; The structure of a Compiler; The evolution pf programming languages; The science of building a Compiler; Applications of compiler technology; Programming language basics.
Lexical analysis: The Role of Lexical Analyzer; Input Buffering; Specifications of Tokens; Recognition of Tokens.

UNIT – 2   6 Hours
Syntax Analysis – 1: Introduction; Context-free Grammars; Writing a Grammar. Top-down Parsing; Bottom-up Parsing.

UNIT – 3   6 Hours
Syntax Analysis – 2: Top-down Parsing; Bottom-up Parsing.

UNIT – 4   6 Hours
Syntax Analysis – 3: Introduction to LR Parsing: Simple LR; More powerful LR parsers (excluding Efficient construction and compaction of parsing tables) ; Using ambiguous grammars; Parser Generators.

PART – B

UNIT – 5   7 Hours
Syntax-Directed Translation: Syntax-directed definitions; Evaluation orders for SDDs; Applications of syntax-directed translation; Syntax-directed translation schemes.

UNIT – 6   6 Hours
Intermediate Code Generation: Variants of syntax trees; Three-address code; Translation of expressions; Control flow; Back patching; Switch-statements; Procedure calls.

UNIT – 7 — 6 Hours
Run-Time Environments: Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management; Introduction to garbage collection.

UNIT – 8 — 7 Hours
Code Generation: Issues in the design of Code Generator; The Target Language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks; A Simple Code Generator

Text Books:
1. Alfred V Aho, Monica S.Lam, Ravi Sethi, Jeffrey D Ullman: Compilers- Principles, Techniques and Tools, 2nd Edition, Pearson Education, 2007.
   (Chapters 1, 3.1 to 3.4, 4 excluding 4.7.5 and 4.7.6, 5.1 to 5.4, 6.1, 6.2, 6.4, 6.6, 6.7 to 6.9, 7.1 to 7.5, 8.1 to 8.6.)

Reference Books:
1. Charles N. Fischer, Richard J. leBlanc, Jr.: Crafting a Compiler with C, Pearson Education, 1991.
2. Andrew W Apple: Modern Compiler Implementation in C, Cambridge University Press, 1997.
3. Kenneth C Louden: Compiler Construction Principles & Practice, Cengage Learning, 1997.

DATA COMPRESSION

Subject Code: 10IS663   I.A. Marks : 25
Hours/Week : 04   Exam Hours: 03
Total Hours : 52   Exam Marks: 100

PART – A

UNIT –1 — 7 Hours
Introduction, Lossless Compression -1: Compression techniques; Modeling and coding.
Mathematical preliminaries for lossless compression: Overview; Basic concepts of Information Theory; Models; Coding; Algorithmic information theory; Minimum description length principle.
Huffman coding: Overview; The Huffman coding algorithm, Minimumvariance Huffman codes; Application of Huffman coding for text compression.

60

Rahul

H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

UNIT – 2
6 Hours

Lossless Compression – 2: Dictionary Techniques: Overview; Introduction; Static dictionary; Adaptive dictionary; Applications: UNIX compress, GIF, PNG, V.42.

Lossless image compression: Overview; Introduction; Basics; CALIC; JPEG-LS; Multiresoution approaches; Facsimile encoding: Run-length coding, T.4 and T.6.

UNIT – 3
6 Hours

Basics of Lossy Coding: Some mathematical concepts: Overview; Introduction; Distortion criteria; Models.

Scalar quantization: Overview; Introduction; The quantization problem; Uniform quantizer; Adaptive quantization.

UNIT – 4
7 Hours

Vector Quantization, Differential Encoding: Vector quantization: Overview; Introduction; Advantages of vector quantization over scalar quantization; The LBG algorithm.

Differential Encoding: Overview; Introduction; The basic algorithm; Prediction in DPCM; Adaptive DPCM; Delta modulation; Speech coding; Image coding.

PART - B

UNIT – 5
7 Hours

Some Mathematical Concepts, Transform coding: Some mathematical concepts: Linear systems; Sampling; Discrete Fourier transform; Z-transform.

Transform coding: Overview; introduction; The transform; Transforms of interest; Quantization and coding for transform coefficients; Application to image compression – JPEG; Application to audio compression – MDCT.

UNIT – 6
6 Hours

Subband Coding, Audio Coding: Subband Coding: Overview; introduction; Filters; The basic subband coding algorithm; Bit allocation; Application to speech coding – G.722; Application to audio coding – MPEG audio; Application to image compression.

Audio Coding: Overview; Introduction; MPEG audio coding; MPEG advanced audio coding; Dolby AC3; Other standards.

UNIT – 7   6 Hours
Wavelet-Based Compression: Overview; Introduction; Wavelets; Multiresolution and the scaling function; Implementation using Filters; Image compression; Embedded zerotree coder; Set partitioning in hierarchical trees; JPEG 2000.

UNIT – 8   7 Hours
Video Compression: Overview; Introduction; Motion compensation; Video signal representation; H.261; Model-based coding; Asymmetric applications; MPEG-1 and MPEG-2; H.263; H.264, MPEG-4 and advanced video coding; Packet video.

Text Books:
1. Khalid Sayood: Introduction to Data Compression, 3rd Edition, Elsevier, 2006. (Chapters 1, 2 excluding 2.2.1 and 2.4.3, 3.1, 3.2, 3.2.1, 3.8.2, 5, 7.1 to 7.5, 7.6, 7.6.1, 7.6.2, 8.1 to 8.3, 8.6, 9.1 to 9.5, 10.1 to 10.4, 11, 12.6 to 12.9, 13, 14.1 to 14.4, 14.9 to 14.12, 15, 16, 18.1 to 18.13)
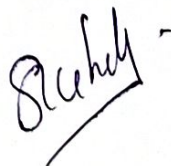
Reference Books:
1. D. Salomon: Data Compression: The Complete Reference, Springer, 1998.

PATTERN RECOGNITION

Subject Code: 10IS664	I.A. Marks : 25
Hours/Week : 04	Exam Hours: 03
Total Hours : 52	Exam Marks: 100

PART – A

UNIT – 1   6 Hours
Introduction: Machine perception, an example; Pattern Recognition System; The Design Cycle; Learning and Adaptation.

UNIT – 2   7 Hours
Bayesian Decision Theory: Introduction, Bayesian Decision Theory; Continuous Features, Minimum error rate, classification, classifiers, discriminant functions, and decision surfaces; The normal density; Discriminant functions for the normal density.

UNIT – 3   7 Hours
Maximum-likelihood and Bayesian Parameter Estimation: Introduction; Maximum-likelihood estimation; Bayesian Estimation; Bayesian parameter estimation: Gaussian Case, general theory; Hidden Markov Models.

UNIT – 4   6 Hours
Non-parametric Techniques: Introduction; Density Estimation; Parzen windows; k_n – Nearest- Neighbor Estimation; The Nearest- Neighbor Rule; Metrics and Nearest-Neighbor Classification.

PART – B

UNIT – 5   7 Hours
Linear Discriminant Functions: Introduction; Linear Discriminant Functions and Decision Surfaces; Generalized Linear Discriminant Functions; The Two-Category Linearly Separable case; Minimizing the Perception Criterion Functions; Relaxation Procedures; Non-separable Behavior; Minimum Squared-Error procedures; The Ho-Kashyap procedures.

UNIT – 6   6 Hours
Stochastic Methods: Introduction; Stochastic Search; Boltzmann Learning; Boltzmann Networks and Graphical Models; Evolutionary Methods.

UNIT – 7   6 Hours
Non-Metric Methods: Introduction; Decision Trees; CART; Other Tree Methods; Recognition with Strings; Grammatical Methods.

UNIT – 8   7 Hours
Unsupervised Learning and Clustering: Introduction; Mixture Densities and Identifiability; Maximum-Likelihood Estimates; Application to Normal Mixtures; Unsupervised Bayesian Learning; Data Description and Clustering; Criterion Functions for Clustering.

Text Books:
1. Richard O. Duda, Peter E. Hart, and David G.Stork: Pattern Classification, 2nd Edition, Wiley-Interscience, 2001.

Reference Books:
1. Earl Gose, Richard Johnsonbaugh, Steve Jost: Pattern Recognition and Image Analysis, PHI, Indian Reprint 2008.

63

[signature]
H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

COMPUTER GRAPHICS AND VISUALIZATION

Subject Code: 10IS665   I.A. Marks : 25
Hours/Week : 04   Exam Hours: 03
Total Hours : 52   Exam Marks: 100

PART - A

UNIT – 1   7 Hours
Introduction: Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging Systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable Pipelines; Performance Characteristics
Graphics Programming: The Sierpinski gasket; Programming Two Dimensional Applications.

UNIT – 2   6 Hours
The OpenGL: The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program; Polygons and recursion; The three-dimensional gasket; Plotting Implicit Functions

UNIT – 3   7 Hours
Input and Interaction: Interaction; Input devices; Clients and Servers; Display Lists; Display Lists and Modeling; Programming Event Driven Input; Menus; Picking; A simple CAD program; Building Interactive Models; Animating Interactive Programs; Design of Interactive Programs; Logic Operations

UNIT – 4   6 Hours
Geometric Objects and Transformations-I: Scalars, Points, and Vectors; Three-dimensional Primitives; Coordinate Systems and Frames; Modeling a Colored Cube; Affine Transformations; Rotation, Translation and Scaling;

PART - B

UNIT – 5   5 Hours
Geometric Objects and Transformations-II: Geometric Objects and Transformations; Transformation in Homogeneous Coordinates; Concatenation of Transformations; OpenGL Transformation Matrices; Interfaces to three-dimensional applications; Quaternion's.

UNIT – 6   7 Hours
Viewing: Classical and computer viewing; Viewing with a Computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden-

64

surface removal; Interactive Mesh Displays; Parallel-projection matrices; Perspective-projection matrices; Projections and Shadows.

UNIT – 7　　　　　　　　　　　　　　　　　　　　　　6 Hours
Lighting and Shading: Light and Matter; Light Sources; The Phong Lighting model; Computation of vectors; Polygonal Shading; Approximation of a sphere by recursive subdivisions; Light sources in OpenGL; Specification of materials in OpenGL; Shading of the sphere model; Global Illumination.

UNIT – 8　　　　　　　　　　　　　　　　　　　　　　8 Hours
Implementation: Basic Implementation Strategies; Four major tasks; Clipping; Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon Rasterization; Hidden-surface removal; Antialiasing; Display considerations.

Text Books:
1. Edward Angel: Interactive Computer Graphics A Top-Down Approach with OpenGL, 5th Edition, Pearson Education, 2008. (Chapters 1 to 7)

Reference Books:
1. Donald Hearn and Pauline Baker: Computer Graphics- OpenGL Version, 3rd Edition, Pearson Education, 2004.
2. F.S. Hill Jr.: Computer Graphics Using OpenGL, 3rd Edition, PHI, 2009.
3. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Pearson Education 1997.

PROGRAMMING LANGUAGES

Subject Code: 10IS666	**I.A. Marks : 25**
Hours/Week : 04	**Exam Hours: 03**
Total Hours : 52	**Exam Marks: 100**

PART - A

UNIT – 1　　　　　　　　　　　　　　　　　　　　　　7 Hours
Introduction; Names, Scopes, and Bindings: The art of language design; Programming language spectrum; Why study programming languages? Compilation and interpretation; Programming environments.
Names, scope, and bindings: The notion of binding time; Object lifetime and storage management; Scope rules; Implementing scope; The meaning of names within a scope; The binding of referencing environments; Macro expansion.

65

UNIT – 2
7 Hours

Control Flow: Expression evaluation; Structured and unstructured flow; Sequencing; Selection; Iteration; Recursion; Non-determinacy

UNIT – 3
6 Hours

Data Types: Type systems; Type checking; Records and variants; Arrays; Strings; Sets; Pointers and recursive types; Lists; Files and Input/Output; Equality testing and assignment.

UNIT – 4
6 Hours

Subroutines and Control Abstraction: Review of stack layout; Calling sequences; Parameter passing; Generic subroutines and modules; Exception handling; Coroutines; Events.

PART – B

UNIT – 5
6 Hours

Data Abstraction and Object Orientation: Object oriented programming; Encapsulation and Inheritance; Initialization and finalization; Dynamic method binding; Multiple inheritance; Object oriented programming revisited.

UNIT – 6
7 Hours

Functional Languages, and Logic Languages: Functional Languages: Origins; Concepts; A review/overview of scheme; Evaluation order revisited; Higher-order functions; Functional programming in perspective. Logic Languages: Concepts; Prolog; Logic programming in perspective.

UNIT – 7
6 Hours

Concurrency: Background and motivation; Concurrency programming fundamentals; Implementing synchronization; Language-level mechanisms; Message passing.

UNIT – 8
7 Hours

Run-Time Program Management: Virtual machines; Late binding of machine code; Inspection/introspection.

Text Books:

1. Michael L. Scott: Programming Language Pragmatics, 3rd Edition, Elsevier, 2009.
   (Chapters 1.1 to 1.5, 3.1 to 3.7, 6 excluding the sections on CD, 7 excluding the ML type system, 8, 9, 10 excluding the sections on CD, 11 excluding the sections on CD, 12, 15. Note: Text Boxes titled Design & Implementation are excluded)

66

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

Reference Books:
1. Ravi Sethi: Programming languages Concepts and Constructs, 2nd Edition, Pearson Education, 1996.
2. R Sebesta: Concepts of Programming Languages, 8th Edition, Pearson Education, 2008.
3. Allen Tucker, Robert Nonan: Programming Languages, Principles and Paradigms, 2nd Edition, Tata McGraw-Hill, 2007.

FILE STRUCTURES LABORATORY

Subject Code: 10ISL67   I.A. Marks : 25
Hours/Week : 03   Exam Hours: 03
Total Hours : 42   Exam Marks: 50

PART - A

Design, develop, and implement the following programs

1. Write a C++ program to read series of names, one per line, from standard input and write these names spelled in reverse order to the standard output using I/O redirection and pipes. Repeat the exercise using an input file specified by the user instead of the standard input and using an output file specified by the user instead of the standard output.

2. Write a C++ program to read and write student objects with fixed-length records and the fields delimited by "|". Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

3. Write a C++ program to read and write student objects with Variable - Length records using any suitable record structure. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.

4. Write a C++ program to write student objects with Variable - Length records using any suitable record structure and to read from this file a student record using RRN.

5. Write a C++ program to implement simple index on primary key for a file of student objects. Implement add ( ), search ( ), delete ( ) using the index.

6. Write a C++ program to implement index on secondary key, the name, for a file of student objects. Implement add ( ), search ( ), delete ( ) using the secondary index.

7. Write a C++ program to read two lists of names and then match the names in the two lists using Cosequential Match based on a single loop. Output the names common to both the lists.

8. Write a C++ program to read k Lists of names and merge them using k-way merge algorithm with k = 8.

9. Write a C++ program to implement B-Tree for a given set of integers and its operations insert ( ) and search ( ). Display the tree.

10. Write a C++ program to implement B+ tree for a given set of integers and its operations insert ( ), and search ( ). Display the tree.

11. Write a C++ program to store and retrieve student data from file using hashing. Use any collision resolution technique.

12. Write a C++ program to reclaim the free space resulting from the deletion of records using linked lists.

Note: In the examination *each* student picks one question from the lot of *all* 12 questions.

SOFTWARE TESTING LABORATORY

Subject Code: 10ISL68   I.A. Marks : 25
Hours/Week : 03   Exam Hours: 03
Total Hours : 42   Exam Marks: 50

1. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Derive test cases for your program based on decision-table approach, execute the test cases and discuss the results.

2. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on boundary-value analysis, execute the test cases and discuss the results.

68

H. O. D.
Dept. Of Information Science & Engineering
Aiva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

3. Design and develop a program in a language of your choice to solve the triangle problem defined as follows: Accept three integers which are supposed to be the three sides of a triangle and determine if the three values represent an equilateral triangle, isosceles triangle, scalene triangle, or they do not form a triangle at all. Assume that the upper limit for the size of any side is 10. Derive test cases for your program based on equivalence class partitioning, execute the test cases and discuss the results.

4. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of dataflow testing, derive different test cases, execute these test cases and discuss the test results.

5. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.

6. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of equivalence class testing, derive different test cases, execute these test cases and discuss the test results.

7. Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of decision table-based testing, derive different test cases, execute these test cases and discuss the test results.

8. Design, develop, code and run the program in any suitable language to implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

9. Design, develop, code and run the program in any suitable language to implement the quicksort algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

10. Design, develop, code and run the program in any suitable language to implement an absolute letter grading procedure, making suitable assumptions. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

11. Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of boundary value testing, derive different test cases, execute these test cases and discuss the test results.

12. Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of equivalence class value testing, derive different test cases, execute these test cases and discuss the test results.

69

H. O. D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225