B.S.ABDUR RAHMAN
U N I V E R S I T Y
B.S.ABDUR RAHMAN INSTITUTE OF SCIENCE & TECHNOLOGY
(Estd.u/s 3 of the UGC Act, 1956)
(FORMERLY B.S.ABDUR RAHMAN CRESCENT ENGINEERING COLLEGE)
Seethakathi Estate, G.S.T. Road, Vandalur, Chennai - 600 048.

B.S ABDUR RAHMAN UNIVERSITY, CHENNAI – 48
POST GRADUATE PROGRAMMES
CREDIT BASED CURRICULUM AND EVALUATION SYSTEM

B.S ABDUR RAHMAN UNIVERSITY, CHENNAI – 48
REGULATIONS -2009 FOR
M.TECH / MCA / M. Sc DEGREE PROGRAMMES

1.0 Preamble

The aim of postgraduate education programme is to be impart in depth knowledge and skill in specialised area pertaining to a discipline to meet the needs of industry, educational and R&D institutions. Due to continuous development coupled with varied needs of industry, the curriculum and syllabi requires to be updated frequently. Moreover the caliber of students admitted to the P.G. programmes also varies widely. Hence, as a deemed to be University there is an imperative need for adoption of flexible system of curriculum design so as to improve the teaching learning process.

Many autonomous Institutions of higher learning like I.I.T.'s, unitary universities like Anna University and Autonomous Colleges have identified the semester system pattern with Credit based Curriculum and continuous assessment as the best system both for under-graduate and post-graduate programmes. Credit based system is a flexible curriculum pattern with many merits. It is devoid of many limitations associated with the conventional rigid pattern of curriculum. Most of the U.S Universities follow the credit based systems.

2.0 The credit based system

i) The credit system is a flexible system of curriculum design in which any academic programme is designed with the required Courses (subjects) of study. Each course is assigned a weight (credit) depending upon its relative importance to the programme of definite total credit rating. A course of study comprises in general, lectures, tutorials, practices, seminar, training etc. Credit assigned to a course essentially depends on the contact periods / week for that course and on the intensity of knowledge transfer / learning.

The credit pattern suggested for adoption is as follows:

a) Every lecture & Tutorial period per week : 1 credit

b) Every laboratory / practice of 2-3 period per week : 1 credit

c) Four weeks of practical training : 1 credit

d) Project work of 12 periods : 6 credits

e) Project work of one full semester : 18 credits

ii) The student's performance in each course is evaluated through continuous assessment which is carried out at different points in time during the semester and is reflected in a grade. There is a grade point associated with each grade. The student earns the credits assigned to a course if he / she secures atleast the minimum pass grade. Otherwise he / she has to re-register for this course if it is a Core course.

iii) The students performance in a semester is measured using Grade point Average (GPA). The GPA is defined as

$$GPA = \frac{\sum_{i}^{n} C_i \; GP_i}{\sum_{i}^{n} C_i}$$

where C_i = the credit for ith course

GP_i = Grade Point secured in the ith course

n = No. of courses registered by the candidate in that semester

iv) The student's performance at any point in time in the programme is measured using the Cumulative Grade Point Average (CGPA) which is defined by a formula similar to the above but the summation over all the courses registered in the various semesters up to that point in time. The overall performance of the student in the programme is also given by the CGPA.

3.0 Curriculum design

i) Curriculum Design is an involved exercise since it needs integration of not only the current educational needs of the specialised profession, but also the anticipated needs arising out of the fast changing national and international technological scene. To make the curriculum both dynamic so as to meet the evolving needs and yet flexible, it is necessary to identify the core part of the curriculum which embodies scientific and engineering knowledge basic to the field. To this core, is added in different proportions, the other ingredients of knowledge of both current and emerging technological processes and systems. With a proper balancing of the core, specialised and elective subjects and suitable integration of meaningful practical and field exercises and challenging

project activity, the curriculum can, not only provide the students with relevant specialised knowledge, but also develop in them the capacity to tackle unknown engineering problems and help them acquire sound professional ethics and an awareness of their obligations of society.

ii) The first step in the curriculum design is to define the credit in terms of number of periods / week of Lecture / Laboratory etc. as indicated supra. Then, the minimum duration of completion of the programme and average work load of student per semester in terms of credits must be fixed. For example, for M.Tech. programme the minimum duration is chosen as 4 semesters and the average load is chosen about 20 credits / semester, so that the total credits to be earned by a student for the award of M.Tech.degree programme is 75 to 80. There can be variation to a limited extent in each semester depending up on the need.

iii) The second step in the Curriculum design is to arrive at the optimal mix of different categories of Courses which are essential to convert a student into a professional engineer. The optimal mix is usually prescribed by means a "Minimum Credit Requirement" for various P.G. programmes is given below:

Programme	**Minimum credit range**
M.Tech	70-80
MCA	100
M.B.A.	90
M.Phil	28-30
M.Sc.	74-80

The above a minimum requirement indicated is based on the experience and prevalent practices

iv) The next step is to list under each of these categories the titles of various courses that are to be taught under the programme. The credit requirement for each course is also fixed. One way of accommodating all these courses is to make the curriculum flexible by identifying certain essential courses as Core Courses and putting the other courses under different elective slots.

v) The final step in the Curriculum Design is to prepare a "Nominal Curriculum" by listing the various courses, both Cores and Electives, to be credited under each semester. Sequencing of the Courses and fixing the pre-requisites are important.

4.0 Features of the credit system

Important feature of the proposed credit based system is given below.

i) The programme is to be completed within the stipulated maximum duration. The minimum and maximum period of the P.G. programmes will be as follows:

Programmes	Min. No. of semesters	Maximum No. of of semesters
M.Tech. (full time)	4	8
M.Tech (part time)	6	12
MCA (full time)	6	12
M.B.A / M.Sc.(full time)	4	8
M.B.A (part time)	6	12
M.Phil.(full time)	2	4

ii) Well before the beginning of every semester each student has to register for the courses to be undertaken in that semester and just before the beginning of the semester the student should enroll, indicating the actual courses registered by him / her.

iii) Each student will be attached to a Faculty Adviser who will provide necessary guidance and help required by the student throughout the programme.

iv) For Lecture based courses a minimum of two sessional assessments will be made besides the End semester Examination and one End semester Examinations for each Laboratory based course.

v) A Class Committee comprising the Course Instructors, Faculty Advisers and Student Representatives monitors the progress of the students and alleviates any difficulties faced in implementation of the academic programme.

vi) Semester load in the nominal curriculum may vary between 15 to 20 credits. However, a student can, off-load some credits in a semester, if required, in consultation with the Faculty Adviser.

vii) Each eligible student of PG programme has to carryout a project work / thesis / dissertation under the supervision of a qualified teacher in the concerned Department during the last semester of the programme.

viii) A student will have to earn a minimum number of total credits prescribed in the curriculum to register for the project semester. The minimum credits specified is given below:

M.Tech (full time)	18 (III semester)
M.Tech (part time)	18 (V semester)
MCA (full time)	45 (VI semester)
M.B.A (full time)	27 (IV semester)
M.B.A (part time)	27 (IV semester)
M.Sc.	28 (IV semester)

ix) If a student does not earn the required credit specified he / she has to complete the arrears to the extent of minimum credit required and then register for the project semester.

5.0 Assessment System

5.1 Continuous Assessment

i) Evaluation of student's performance in individual courses is an Important Component of Teaching – Learning process. Continuous assessment helps not only in monitoring the progress of the student but also in taking corrective action in time to improve the performance of the student. The system of sessional assessments and examinations to be conducted for each lecture-based course offered in the programme will be as follows:

a) The following rule will apply to

All Full Time and Part Time PG. Programmes (M.Tech, M.B.A, MCA and M.Sc.)

For lecture-based courses, normally a minimum of two sessional assessments will be made during the semester. The sessional assessments may be combination of periodical tests and assignments. The assessment procedure as decided at the Class Committee will be announced to the students right at the beginning of the semester by the teacher and informed to the Head of Institution.

M.B.A. – The tests followed by one case study and assignment will be made during the semester.

b) There shall be one end semester examination of 3 hours duration in each lecture based course.

c) The evaluation of the Project work will be based on the project report and a Viva-Voce Examination by a team consisting of the supervisor concerned, an Internal Examiner and External Examiner to be appointed by the Controller of Examinations.

d) At the end of practical training or industrial attachment, the candidate shall submit a certificate from the organization where he / she has undergone training and also a brief report. The evaluation will be made based on this report and a Viva-Voce Examination, conducted internally by a Departmental Committee constituted by the Head of the Department.

e) A student who missed an assessment test including end semester examinations for genuine reasons, may be permitted to write a substitute examination. Such permission can be accorded only under exceptional circumstances such as accident or admission to a hospital due to illness.

ii) In Laboratory-based courses the Instructor will decide the mode of sessional assessment. The Instructor may even go in for evaluation of every experiment conducted by the student. There will be one End-semester examinations of 3 hours in each Lab-based course.

iii) The progress of Project work will be assessed continuously during the semester by a committee of faculty members constituted by the HOD. Atleast minimum of three assessments will be made. The student will have to submit a written progress report duly attested by the guide before he appears before the assessment committee.

5.2 Weightage

The weightages for the various components of assessment are as follows:

a) Lecture-based courses :

Continuous assessment (Equal weightage for each test)	: 50%
End-Semester Examinations	: 50%

b) Laboratory-based courses :

Continuous Assessment	: 75%
End-Semester Examinations	: 25%

c) Project Work :

Continuous Assessment	: 50%
Evaluation of Project report by External Examiners	: 20%
Viva-voce Examinations	: 30%

1.5 Grading System

i) Based on the semester performance, each student is awarded a final letter grade at the end of the semester in each course. The letter grades, the mark ranges and the corresponding grade points are as follows, but grading has to be relative grading:

Mark range (suggested only)	Letter grade	Grade points
90 - 100	S	10
80 - 89	A	9
70 - 79	B	8
60 - 69	C	7
55 - 59	D	6
50 - 54	E	5
0 - 49	U	0
Incomplete	I	-
Withdrawal	W	-

Relative, flexible grading system will be adopted

"W" denotes withdrawal from the course

"I" denotes inadequate attendance and hence prevention from End-semester examination.

"U" denotes failure in a course.

ii) A student is considered to have completed a course successfully and earned the credits if he / she secure 5 grade points or higher. A letter grade U in any course implies a failure in that course. A course successfully completed cannot be repeated for the purpose of improving Cumulative Grade Points Average.

iii) A final meeting of the Class Committee without the student members(s) will be convened within ten days after the last day of the end-semester examination. The letter grades to be awarded to the students for different courses will be finalized at the meeting.

7.0. Course Codification

For all core and laboratory course, a five digit alpho-numerical code has been adopted. The first two digits represent the Department offering the subject and the next three digits indicate the serial number of the course. For the elective subjects a six digit code adopted. The first two digits represent the Department offering the elective, the third digit 'Y' identifies, that it is a postgraduate elective and the last three digits indicate the course serial number.

MASTER OF COMPUTER APPLICATIONS (MCA)
(Six Semesters / Full Time)
(Updated Curriculum and Syllabi as on June 2012)

CURRICULAM

SEMESTER I

Sl.No.	Code	Subject	L	T	P	C	TC
Theory							
1.	MA617	Mathematical Foundations for Computer Applications	3	1	0	4	
2.	CA601	Problem Solving and Programming in C	3	0	0	3	
3.	CA602	Computer Architecture	3	0	0	3	
4.	CA603	Database Management Systems	3	0	0	3	
5.	CA604	Software Engineering	3	0	0	3	
6.	CA605	Computer Networks	3	0	0	3	
Practical							
1.	CA606	C Programming Lab	0	0	6	2	
2.	CA607	DBMS Lab	0	0	6	2	
		Total					**23**

SEMESTER II

Sl.No.	Code	Subject	L	T	P	C	TC
Theory							
1.	MS681	Accounting and Financial Management	3	1	0	4	
2.	MA618	Resource Management Techniques	3	1	0	4	
3.	CA608	Data Structures	3	0	0	3	
4.	CA609	Operating Systems	3	0	0	3	
5.	CA610	Design and Analysis of Algorithms	3	0	0	3	
6.	CA611	Object Oriented Programming using C++	3	0	0	3	
Practical							
1.	CA612	Object Oriented Programming Lab	0	0	6	2	
2.	CA613	Algorithms and Data Structures Lab	0	0	6	2	
3.	EN611	Communication Skills Laboratory	0	0	2	1	
		Total					**25**

SEMESTER III

Sl.No.	Code	Subject	L	T	P	C	TC
Theory							
1.	CA701	Internet and Java Programming	3	0	0	3	
2.	CA702	Computer Graphics and Multimedia Systems	3	0	0	3	
3.	CA703	Microprocessors and its Applications	3	0	0	3	
4.	CA704	Distributed Computing	3	0	0	3	
5.		Elective I	3	0	0	3	
6.		Elective II	3	0	0	3	
Practical							
1.	CA705	Java Programming and Graphics & Multimedia Lab	0	0	6	2	
2.	CA706	Microprocessor Lab	0	0	6	2	
3.	CA713	Soft Skills	0	0	2	1	
		Total					**23**

Semester IV

Sl.No.	Code	Subject	L	T	P	C	TC
Theory							
1.	CA707	Object Oriented Analysis and Design	3	0	0	3	
2.	CA708	Web Graphics	3	0	0	3	
3.	CA709	Unix and Network Programming	3	0	0	3	
4.	CA710	XML and Web Services	3	0	0	3	
5.	Elective III		3	0	0	3	
6.	Elective IV		3	0	0	3	
Practical							
1.	CA711	Unix and Network Programming Lab	0	0	6	2	
2.	CA712	XML and Web Services Lab	0	0	6	2	
		Total					**22**

Semester V

Sl.No.	Code	Subject	L	T	P	C	TC
Theory							
1.	CA801	Principles of Compiler Design	3	0	0	3	
2.	MS084	Organizational Behaviour	3	0	0	3	
3.	CA803	Middleware Technologies	3	0	0	3	
4.		Elective V	3	0	0	3	
5.		Elective VI	3	0	0	3	
Practical							
1.	CA804	Middleware Lab	0	0	6	2	
2.	CA805	Project Phase - I	0	0	6	*	
			Total				**17**

Semester VI

Sl.No.	Code	Subject	L	T	P	C	TC
Practical							
1.	CA806	Project Phase - II	0	0	36	18	
			Total			**18 + * = 24**	

* Phase I credit 6 will be considered in Phase II

Total Credits : 134

LIST OF ELECTIVES

Code	Subject	L	T	P	C	TC
CAY001	Management Information System	3	0	0	3	
CAY002	E-Commerce	3	0	0	3	
CAY003	Advanced Databases	3	0	0	3	
CAY004	Digital Image Processing	3	0	0	3	
CAY005	TCP/IP Protocol Suite	3	0	0	3	
CAY006	Visual Programming	3	0	0	3	
CAY007	Mobile Computing	3	0	0	3	
CAY008	Software Quality Management	3	0	0	3	
CAY009	Data Mining and Data Warehousing	3	0	0	3	
CAY010	Component Based Technology	3	0	0	3	
CAY011	Supply Chain Management	3	0	0	3	
CAY012	Business Processes	3	0	0	3	
CAY013	Software Project Management	3	0	0	3	
CAY014	Enterprise Resource Planning	3	0	0	3	
CAY015	Software Agents	3	0	0	3	
CAY016	Unix Internals	3	0	0	3	
CAY017	Grid Computing	3	0	0	3	
CAY018	Network Security	3	0	0	3	
CAY019	Embedded Systems	3	0	0	3	
CAY020	Software Quality Assurance	3	0	0	3	
CAY021	Adhoc Networks	3	0	0	3	
CAY022	Special Elective	3	0	0	3	

MASTER OF COMPUTER APPLICATIONS (MCA)
(Six Semesters / Full Time)
SYLLABUS

MA617	MATHEMATICAL FOUNDATIONS FOR COMPUTER APPLICATIONS	L	T	P	C
		3	1	0	4

OBJECTIVE OF THE COURSE:

To extend student's mathematical maturity and ability to deal with abstraction and to introduce most of the basic terminologies used in computer science courses and application of ideas to solve practical problems.

Outcomes of the Course:

At the end of the course students would

- have knowledge of the concepts needed to test the logic of a program.
- have gained knowledge which has application in expert system, in database and a basic for the prolog language.
- have an understanding in identifying patterns on many levels.
- be aware of a class of functions which transform a finite set into another finite set which relates to input and output function in computer applications.
- be exposed to concepts and properties of algebraic structures.
- have a basic knowledge to study complier design.

1. MATRIX ALGEBRA **12**

Matrices, Rank of Matrix, Solving System of Equations-Eigen Values and Eigen Vectors-Inverse of a Matrix - Cayley Hamilton Theorem.

2. BASIC SET THEORY **12**

Basic Definitions - Venn Diagrams and set operations - Laws of set theory - Principle of inclusion and exclusion - partitions- Permutation and Combination - Relations- Properties of relations - Matrices of relations - Closure operations on relations - Functions - injective, surjective and bijective functions.

3. MATHEMATICAL LOGIC **12**

Propositions and logical operators - Truth table - Propositions generated by a set, Equivalence and implication - Basic laws- Some more connectives -

Functionally complete set of connectives- Normal forms - Proofs in Propositional calculus - Predicate calculus.

4. FORMAL LANGUAGES **12**

Languages and Grammars-Phrase Structure Grammar-Classification of Grammars-Pumping Lemma For Regular Languages-Context Free Languages.

5. FINITE STATE AUTOMATA **12**

Finite State Automata-Deterministic Finite State Automata(DFA), Non Deterministic Finite State Automata (NFA)-Equivalence of DFA and NFA-Equivalence of NFA and Regular Languages.

L : 45, T : 15
Total : 60

TEXT BOOK :

1. Kenneth H.Rosen, " Discrete Mathematics and Its Applications", Tata McGraw Hill, Fourth Edition, 2002 (Unit 1,2 & 3).

REFERENCES :

1. Hopcroft and Ullman, "Introduction to Automata Theory, Languages and Computation", Narosa Publishing House, Delhi, 2002. (Unit 4,5)
2. A.Tamilarasi & A.M.Natarajan, "Discrete Mathematics and its Application", Khanna publishers, 2nd Edition 2005.
3. M.K.Venkataraman "Engineering Mathematics", Volume II, National Publishing Company, 2nd Edition,1989.

CA601 PROBLEM SOLVING AND PROGRAMMING IN C

L	T	P	C
3	0	0	3

OBJECTIVE OF THE COURSE:

The aim of the course is

- to introduce students to the foundations of computing, programming and problem-solving.
- to develop basic programming skills necessary for engineering education.
- to develop the programming ability in students, and to improve their proficiency in applying the computing fundamentals to their field of study.

Outcomes of the Course:

At the end of the course the student will have strong fundamentals in C.

1. INTRODUCTION TO COMPUTER PROBLEM SOLVING 9

Introduction – The Problem Solving aspect – Top down Design – Implementation of Algorithms – Program Verification – Efficiency of Algorithms – Analysis of Algorithms

2. FUNDAMENTAL ALGORITHMS 9

Introduction – Exchanging the values – Counting – Factorial Computation – SINE computation – Base Conversion – Factoring Methods – Array Techniques.

3. INTRODUCTION TO C LANGUAGE 9

Overview of C – Constants, Variables and Data Types – Operators and Expressions – Managing Input/Output Operations – Formatted I/O – Decision Making - Branching –- IF, Nested IF – Switch – goto - Looping- While, do, for statements.

4. ARRAYS, FUNCTIONS, STRUCTURES AND UNIONS 9

Arrays – dynamic and multi-dimensional arrays - Character arrays and Strings – String handling Functions - User defined Functions – Categories of Functions – Recursion - Structures and Unions – Array of Structures – Structures and Functions

5. POINTERS AND FILE MANAGEMENT **9**

Pointers – Declaration, Accessing a variable, character strings, pointers to functions and structures - File Management in C – Dynamic Memory allocation – Linked Lists – Preprocessors.

Total: 45

TEXT BOOKS :

1. R.G.Dromey " How to Solve it by Computer ", PHI , 1998
2. E.Balagurusamy " Programming in ANSI C " , Tata McGraw Hill, 2004

REFERENCES :

1. Deitel and Deitel " C How to Program ", Addisson Wesley , 2001
2. Brian W.Kernighan & Dennis Ritchie "C Programming Language", PHI, 1990
3. Byron.S.Gottfried "Schaum's Outline of Programming with C ", 2nd Edition,1996

CA602 COMPUTER ARCHITECTURE **L T P C**
3 0 0 3

OBJECTIVE OF THE COURSE:

The aim of the course is to

- understand the digital logic fundamentals
- know about the digital components
- understand the computer organization and Programming concepts
- know in detail I-O organization
- describe about the memory organization and CPU

Outcomes of the Course:

At the end of the course, the students will be able to

- understand how computer hardware has evolved to meet the needs of multi-processing systems.
- understand the major components of a computer including CPU, memory, I/O and storage.
- understand the uses for cache memory.
- understand a wide variety of memory technologies both internal and external.
- understand the role of the operating system in interfacing with the computer hardware.
- understand the basic components of the CPU including the ALU and control unit.
- have a basic understanding of assembly programming.

1. INTRODUCTION TO DIGITAL DESIGN **9**

Data Representation – Data Types – Complements – Arithmetic Operations – Representations – Fixed –Point, Floating – Point , Decimal Fixed – Point – Binary Codes- Logic Gates, Boolean Algebra, Map Simplification – Combinational Circuits: Half-Adder, Full Adder- Flip Flops - Sequential Circuits

2. DIGITAL COMPONENTS REGISTER TRANSFER & MICRO OPERATIONS **9**

ICs – Decoders – Multiplexers – Registers – Shift Registers – Binary Counters – Memory Unit – Register Transfer Language – Register Transfer – Bus And

Memory Transfers – Arithmetic , Logic And Shift Micro Operations , Arithmetic Logic Shift Unit.

3. COMPUTER ARCHITECTURE AND PROGRAMMING 9

Instruction Codes – Computer Registers – Computer Instructions – Timing And Control – Instruction Cycle – Memory Reference Instructions – I/O And Interrupt – Machine Language – Assembly Language – Assembler - Program Loops – Programming Arithmetic And Logic Operations – Subroutines – I/O Programming.

4. INPUT – OUTPUT ARCHITECTURE 9

Peripheral Devices – Input-Output Interface – Asynchronous Data Transfer – Modes Of Transfer – Priority Interrupt – DMA – IOP – Serial Communication.

5. MEMORY ARCHITECTURE AND CPU 9

Memory Hierarchy – Main Memory – Auxiliary Memory – Associative Memory – Cache Memory – Virtual Memory – Memory Management Hardware – CPU: General Register Organization – Control Word – Stack Organization – Instruction Format – Addressing Modes – Data Transfer And Manipulation – Program Control.

Total : 45

TEXTBOOK :

1. M.Morris Mano,"Computer System Architecture",Prentice Hall of India, 2001.

REFERNCES :

1. John .p.Hayes,"Computer Architecture and Organization", Tata McGraw Hill, 1996.
2. V.C.Hamatcher, et al "Computer Organization", Tata Mcgraw Hill,1996

CA603 DATABASE MANAGEMENT SYSTEMS

L	T	P	C
3	0	0	3

OBJECTIVE OF THE COURSE:

The course, Database Management Systems, provides an introduction to the management of database systems. The course emphasizes the understanding of the fundamentals of relational systems including data models, database architectures, and database manipulations. The course uses a problem-based approach to learning.

Outcomes of the Course:

At the conclusion of the course, the student will be able to:

- understand terms related to database design and management
- understand the objectives of data and information management
- understand the database development process
- understand the relational model and relational database management system
- assess data and information requirements
- construct conceptual data models
- develop logical data models
- evaluate the normality of a logical data model, and correct any anomalies
- develop physical data models for relational database management systems
- implement relational databases using a RDBMS
- retrieve data using SQL

1. INTRODUCTION **9**

Database Systems vs. File Systems-View of Data- Data Models-Database Languages-Transaction Management-Database Systems Structure-History of Database Systems-Database Systems Applications-Entity Relationship Model

2. RELATIONAL DATABASES **9**

SQL-Basic Structure-Set Operations-Complex Queries-Joined Queries-DDL-Embedded SQL-Dynamic SQL-Other SQL Functions-Query by Example-Integrity and Security of searching-Relational Database Design

3. DATA STORAGE AND INDEXING **9**

Storage & File Structure-Disks-RAID-File Organization-Indexing &Hashing-B+ TREE-B Tree-Static Hashing-Dynamic Hashing-Multiple Key Access

4. QUERY EVALUATION & OPTIMIZATION **9**

Query Processing-Selection Operation-Sorting-Join Operation-Evaluation of Expressions-Query Optimization

5. TRANSACTION MANAGEMENT **9**

Transaction Concept-Static Implementation-Concurrency Control-Protocols-Deadlock Handling-Recovery Systems-Recovery with Concurrent Transactions-Shadow Paging-Buffer Management-Case Studies-Oracle-Microsoft SQL Server

Total: 45

TEXT BOOK:

1. Abraham Silberschatz, Hentry F.Korth and S.Sudharssan,"Database System Concepts", 4th Edition, Tata McGraw Hill, 2002.

REFERENCE :

1. Raghu Ramakrishnan & Johannesgerhrke, "Data Base Management Systems", Mc Graw Hill International Edition, 2000

CA604	SOFTWARE ENGINEERING	L	T	P	C
		3	0	0	3

OBJECTIVE OF THE COURSE:

The aims of the course are as follows

- To define software engineering and explain its importance
- To discuss the concepts of software products and software processes
- To explain the importance of process visibility
- To use appropriate computer science and mathematics principles in the development of software systems.
- To solve problems in a team environment through effective use of written and oral communication skills.
- To have knowledge of current issues presently involved in effectively performing duties as a software practitioner in an ethical and professional manner for the benefit of society.
- To practice the lifelong learning needed in order to keep current as new issues emerge.
- To develop software in at least one application domain.

Outcomes of the Course:

At the completion of this course, students will be able to,

- Understand and apply mathematics, physical science, computer science and related disciplines.
- Understand and apply foundations of software engineering practice and process within production constraints
- Analyze, document and track system requirements
- Design, implement and maintain software systems
- Verify and validate software systems
- Get an awareness of current industry standards and practices
- Work in small, medium and large scale application domains as well as critical application domains
- Understand and apply principles of project management for single, paired, and team processes

- Have strong oral and written communication skills to help students in preparing good quality documentation
- Understand professional responsibility and the application of ethical principles
- Have knowledge of economics of software engineering and the effect of software engineering on the development of the society
- Apply software engineering theory, principles, tools and processes, as well as the theory and principles of computer science and mathematics, to the development and maintenance of complex software systems
- Design and experiment with software prototypes
- Select and use software metrics
- Participate productively on software project teams involving students from both software engineering and other majors
- Have effective communications skills through oral and written reports and software documentation evaluated by both peers and faculty
- Evaluate the business and impact of potential solutions to software engineering problems in a global society, using their knowledge of contemporary issues
- Explain the impact of globalization on computing and software engineering
- Apply appropriate codes of ethics and professional conduct to the solution of software engineering problems
- Identify resources for determining legal and ethical practices in other countries as they apply to computing and software engineering

1. INTRODUCTION **9**

A Generic View of Process – Process Models-The Waterfall Model-Incremental Model-Evolutionary Model-Specialized Model-The Unified Process–Agile Process – Agile Models – Software Cost Estimation – Planning – Risk Analysis – Software Project Scheduling.

2. REQUIREMENT ANALYSIS **9**

System Engineering Hierarchy – System Modeling – Requirements Engineering: Tasks- Initiating The Process-Eliciting Requirements-Developing Use Cases-Negotiating Requirements-Validating Requirements – Building the Analysis Models: Concepts

3. SOFTWARE DESIGN **9**

Design Concepts – Design Models – Pattern Based Design – Architectural Design – Component Level Design – Component – Class Based And Conventional Components Design – User Interface – Analysis And Design.

4. SOFTWARE TESTING **9**

Software Testing – Strategies: Conventional - Object Oriented – Validation Testing – Criteria – Alpha – Beta Testing- System Testing – Recovery – Security – Stress – Performance - Testing Tactics – Testing Fundamentals-Black Box – White Box – Basis Path-Control Structure.

5. SCM AND QUALITY ASSURANCE **9**

Software Configuration And Management-Features-SCM Process-Software Quality Concepts – Quality Assurance – Software Review–Technical Reviews – Formal Approach To Software Quality Assurance – Reliability – Quality Standards – Software Quality Assurance Plan

Total: 45

TEXT BOOK :

1. Roger Pressman.S., “Software Engineering: A Practitioner's Approach”, 6th Edition, Mcgraw Hill, 2005.

REFERENCES :

1. P. Fleeger, “Software Engineering”, Prentice Hall, 1999.
2. Carlo Ghezzi, Mehdi Jazayari, Dino Mandrioli, “Fundamentals Of Software Engineering”, Prentice Hall Of India 1991.
3. I. Sommerville, “Software Engineering” , 5th Edition: Addison Wesley, 1996.

CA605 COMPUTER NETWORKS **L T P C**
3 0 0 3

OBJECTIVE OF THE COURSE:

The intent of this course is to provide students with enough knowledge in networking, various types of networks and to analyze different network applications. It discusses the Network Architecture and defines the quantitative performance metrics that often drive network design. This course explains the overall end-to-end behavior of networks. It introduces internetworking and describes the key elements of the Internet Protocol. It describes many of the issues that all data link protocols must address, including encoding, framing, and error detection. It defines how networks are scaled to the size of the Internet and are able to route packets. It includes core concepts, existing protocol specifications, real protocols which are used to illustrate most of the important ideas.

OUTCOMES OF THE COURSE:

At the end of the course, the students will be able to

- Have good knowledge in Computer Networks
- Identify and describe the layers of the OSI and TCP/IP
- Illustrate how networks work in practice
- Analyze the applications of network
- Make effective use of networking topologies
- Utilize various networking devices efficiently

1. INTRODUCTION **9**

Building a network – Requirements – Network Architecture – OSI – Internet – Direct Link Networks – Hardware building blocks – Framing – Error detection – Reliable transmission.

2. NETWORK FUNDAMENTALS **9**

LAN Technology – LAN Architecture – BUS/Tree – Ring – Star – Ethernet – Token Rings – Wireless.

3. NETWORK LAYER **9**

Packet Switching – Switching and Forwarding – Bridges and LAN switches – Internetworking – Simple Internetworking – Routing.

4. TRANSPORT LAYER **9**

Reliable Byte Stream (TCP) – Simple Demultiplexer (UDP) – TCP Congestion Control – Congestion Avoidance Mechanisms.

5. PRESENTATION LAYER and APPLICATIONS **9**

Presentation formatting – Data compression – Cryptographic Algorithms: RSA - DES — Applications – Domain Name Service – Email - SMTP – MIME – HTTP – SNMP.

Total: 45

TEXT BOOK :

1. Larry L. Peterson & Bruce S. Davie, “Computer Networks - A systems Approach”, 2nd Edition, Harcourt Asia/Morgan Kaufmann, 2000.

REFERENCES :

1. James F. Kurose and Keith W. Ross, “Computer Networking - A Top Down Approach featuring the Internet”, 1st Edition, Addison Wesley Publishing Company, 2001.
2. William Stallings, “Data and Computer Communications”, 5th Edition, PHI, 1997.
3. Andrew S. Tanenbaum, “Computer Networks”, Tata Mcgraw Hill, 3rd Edition, 2001

CA606	C PROGRAMMING LAB	L	T	P	C
		0	0	6	2

OBJECTIVE OF THE COURSE:

The aim of the course is

- to impart knowledge on the fundamentals of C Programming
- to brief on Data Types, Operators, Statements, Loops, Functions, Array, Pointers, Structures

Outcomes of the Course:

At the end of the course, the students will learn the fundamentals of C Programming

1. Display the following:

(i) Floyd's triangle (ii) Pascal Triangle

2. Generate the following series of numbers:

Armstrong numbers between 1 to 100

Prime numbers between 1 to 50

Fibonacci series up to N numbers

3. Manipulate the strings with following operations.

(i) Concatenating two strings (ii) Reversing the string (iii) Finding the substring

(iv) Replacing a string (v) Finding length of the string

4. Find the summation of the following series:

(i) Sine (ii) Cosine (iii) Exponential

5. Create the sales report for M sales person and N products using two dimensional array.

6. Simulate following Banking operations using functions.

(i) Deposit (ii) Withdrawal (iii) Balance Enquiry

7. Implement using recursion

I, Find the solution of Towers of Hanoi problem using recursion.

II, Fibonacci number generation.

III, Factorial

8. Generate Student mark sheets using structures.

9. Create a collection of books using arrays of structures and do the following:
(i) Search a book with title and author name (ii) Sorts the books on title.

CA607	DBMS LAB	L	T	P	C
		0	0	6	2

Objective of the Course:

The aim of the course is

- To learn SQL (Structured Query Language) which would provide functionality to:
- Learn how to create tables which are fundamental storage blocks of data.
- Learn how to place constraints on data that is entered on tables to ensure data integrity.
- Learn how to add, change and remove data from tables.
- Learn how to select a subset of the data you want to see from the collection of tables and data.
- Learn how to combine table and group multiple rows of data in table.
- To learn PL/SQL which would provide the ability to do iterative programming at database level to:
- Write programming blocks with conditional structure, assignment structure, loop structure, etc.
- Use exception Handling, Transaction oriented programs, Stored procedures, functions, packages, etc.
- Use cursors which would allow row wise access of data.
- Use triggers which would allow you define pre and post actions when something changes in the database tables.

Outcomes of the Course:

At the end of this course, a student will be able to

- Get an idea about database and how this is widely used in real time business activities.
- Create and do manipulation on table.
- Work with table by using PL/SQL Program.
- Get a clear idea about database transaction activities.

1. Execute a single line and group functions for a table.
2. Execute DCL and TCL Commands.
3. Create and manipulate various DB objects for a table.
4. Create views, partitions and locks for a particular DB.
5. Write PL/SQL procedure for an application using exception handling.
6. Write PL/SQL procedure for an application using cursors.
7. Write a DBMS program to prepare reports for an application using functions.
8. Write a PL/SQL block for transaction operations of a typical application using triggers.
9. Write a PL/SQL block for transaction operations of a typical application using package.
10. Design and develop an application using any front end and back end tool (make use of ER diagram and DFD).

Typical Applications – Banking, Electricity Billing, Library Operation, Pay roll, Insurance, Inventory, etc.

SEMESTER – II

		L	T	P	C
MS681	**ACCOUNTING AND FINANCIAL MANAGEMENT**	**3**	**1**	**0**	**4**

OBJECTIVE OF THE COURSE:

Finance has rightly been termed as the "Master Key" providing access to all resources required for running business activities. Hence efficient management of business enterprises is closely linked with the efficient management of their finances. In view of the growing importance of the finance function, Financial Management and Accounting plays an important role in any organization.

Outcomes of the Course:

At the end of this course, a student will be able to

- Prepare final accounts of a concern to find out the profit or loss
- Prepare fund and cash flow statements regarding uses and applications of funds
- Analyse the firm by applying various ratios.
- Prepare various budgets
- Take investment decisions
- Analyse the factors affecting a capital structure, working capital and dividends.

1. FINANCIAL ACCOUNTING **12**

Meaning and Scope of Accounting-Principles-Concepts-Conventions-Accounting Standards-Final Accounts-Trial Balance-Trading Account-Profit and Loss Account-Balance Sheet.

2. COST ACCOUNTING **12**

Meaning-Objectives-Elements of Cost-Cost Sheet-Marginal Costing and Cost Volume Profit Analysis-Break Even Analysis-Applications-Limitations

3. MANAGEMENT ACCOUNTING **12**

Budgets and Budgetary Control-Meaning-Types-Sales Budget-Production Budget- Budget-Flexible Budgeting-Cash Budget- Computerized Accounting - Accounting Ratios Analysis-Funds Flow Analysis-Cash Flow Analysis.

4. INVESTMENT DECISION 12

Objectives and Functions of Financial Management-Foreign exchange(Forex)-Risk-Return Relationship-Time Value of Money Concepts-Capital Budgeting-Methods of Appraisal.

5. FINANCING DECISION AND WORKING CAPITAL MANAGEMENT 12

Capital Structure-Factors Affecting Capital Structure-Dividend Policy-Types of Dividend Policy- Concepts of Working Capital-Working Capital Policies-Factors affecting Working Capital-Estimation of Working Capital Requirements

L : 45, T : 15
Total: 60

TEXTBOOKS :

1. S.N.Maheswari, “Financial and Management Accounting”, Sultan Chand & Sons, 2003

2. I.M.Pandey, ”Financial Management”, Vikas Publications, 4th Reprint, 2002

REFERENCES :

1. S.P.Iyengar, “Cost and Management Accounting”, Sultan Chand & Co.

2. I.M.Pandey, “Elements of Management Accounting” Vikas Publishing House, 1999

MA618 RESOURCE MANAGEMENT TECHNIQUES

L	T	P	C
3	1	0	4

OBJECTIVE OF THE COURSE:

To teach students how to solve a real world problem by

- Constructing a model representation of the problem under study.
- Deriving a solution from the model.
- Testing the model and solution
- Implementing the solution.

Outcomes of the Course:

Resource Management Techniques gives students the power to make more effective decisions and build more productive systems based on:

- Data
- Consideration of all available options
- Careful predictions of outcomes and estimates of risk
- The latest decision tools and techniques

1. LINEAR PROGRAMMING MODELS **12**

Mathematical Formulation - Graphical Solution of linear programming models – Simplex method – Artificial variable Techniques- Variants of Simplex method

2. TRANSPORTATION AND ASSIGNMENT MODELS **12**

Mathematical formulation of transportation problem- Methods for finding initial basic feasible solution – optimum solution - degeneracy – Mathematical formulation of assignment models – Hungarian Algorithm – Variants of the Assignment problem.

3. INTEGER PROGRAMMING MODELS **12**

Formulation – Gomory's IPP method – Gomory's mixed integer method – Branch and bound technique.

4. SCHEDULING BY PERT AND CPM **12**

Network Construction – Critical Path Method – Project Evaluation and Review Technique – Resource Analysis in Network Scheduling

5. QUEUEING MODELS **12**

Characteristics of Queuing Models – Poisson Queues - (M / M / 1) : (FIFO / 8 /8), (M / M / 1) : (FIFO / N / 8), (M / M / C) : (FIFO / 8 / 8), (M / M / C) : (FIFO / N / 8) models.

L : 45, T : 15
Total : 60

TEXT BOOK :

1. Taha H.A., "Operations Research : An Introduction " 7th Edition, Pearson Education, 2004.

REFERENCES :

1. A.M.Natarajan, P.Balasubramani, A.Tamilarasi, "Operations Research", Pearson Education, Asia, 2005.

2. Prem Kumar Gupta, D.S. Hira, "Operations Research", S.Chand & Company Ltd, New Delhi, 3rd Edition , 2003.

CA608	DATA STRUCTURES	L	T	P	C
		3	0	0	3

OBJECTIVE OF THE COURSE:

Data structure is a particular way of storing and organizing data in a computer so that it can be used efficiently. Different kinds of data structures are suited to different kinds of applications and some are highly specialized to specific tasks. For example, B-trees are particularly well-suited for implementation of databases, while compiler implementations usually use hash tables to look up identifiers. Data structures are used in almost every program or software system. Data structures provide a means to manage huge amounts of data efficiently, such as large databases and internet indexing services. Usually, efficient data structures are a key to designing efficient algorithms. Some formal design methods and programming languages emphasize data structures rather than algorithms as the key organizing factor in software design.

Outcomes of the Course:

At the completion of this course, the student will able to

- Understand the properties of various data structures;
- Identify the strengths and weaknesses of different data structures concepts;
- Design and employ appropriate data structures for solving computing problems;
- Possess the knowledge of various existing algorithms;
- Analyze and compare the efficiency of algorithms;
- Possess the ability to design efficient algorithms for solving computing problems;

1. DATA STRUCTURES **9**

Introduction – Arrays – Structures – Stack: Definition and examples, Representing Stacks - Queues and lists: Queue and its Representation, lists – Applications of Stack, Queue and Linked Lists.

2. TREES **9**

Binary Trees – Operations on binary trees - Binary Tree Representations – node representation, internal and external nodes, implicit array representation – Binary tree Traversals - Huffman Algorithm – Representing Lists as Binary Trees

3. SORTING AND SEARCHING **9**

General Background – Exchange sorts – Selection and Tree Sorting – Insertion Sorts – Merge and Radix Sorts – Basic Search Techniques – Tree Searching – General Search Trees – Hashing.

4. GRAPHS AND THEIR APPLICATIONS **9**

Graphs – An application of graphs – Representation – transitive closure - Warshall's algorithm – Shortest path algorithm - a flow Problem – Dijkstra's algorithm – An application of scheduling - Linked representation of Graphs – Graph Traversals

5. STORAGE MANAGEMENT **9**

General Lists: Operations, linked list representation, using lists, Freeing list nodes - Automatic list Management: Reference count method, Garbage Collection, Algorithms, Collection and compaction.

Total: 45

TEXT BOOK :

1. Tanaenbaum A.S., Langram Y. Augestein M.J " Data Structures using C" Pearson Education, 2004

REFERENCES:

1. Robert Kruse & Clovis L. Tondo "Data Structures and Program Design in C", Prentice Hall, 2nd edition.,1991.
2. Weiss "Data Structures and Algorithm Analysis in C", Addison Wesley, Second Edition, 1997.

CA609	OPERATING SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVE OF THE COURSE:

This course provides an overview of computer system and operating system, concepts of process management, memory management, storage management, protection and security issues, and distributed systems

Outcomes of the Course:

At the completion of this course, students will be able to

- Master functions, structures and history of operating systems
- Master understanding of design issues associated with operating systems
- Master various process management concepts including scheduling, synchronization, deadlocks
- Understand multithreading
- Master concepts of memory management including virtual memory
- Master system resources sharing among the users
- Master issues related to file system interface and implementation, disk management
- Understand protection and security mechanisms
- Understand various types of operating systems including Unix

1. INTRODUCTION **7**

Definition of OS-Mainframe System-Desktop Systems-Multi processor System-Distributed-Clustered-Real time Systems-Handheld Systems-Operating System Structure-System Components-Services-System Calls-System Programs-System Design and Implementation.

2. PROCESS MANAGEMENT **8**

Concepts-Process Scheduling-Operations on Processes-Co-operating Processes-Inter Process Communication-CPU Scheduling-Scheduling Concepts-Criteria-Scheduling Algorithms-Multiprocessor Scheduling-Real time Scheduling.

3. PROCESS SYNCHRONIZATION **10**

Critical Section-Synchronization Hardware-Semaphores-Problems of Synchronization-Critical Regions-Monitors-Deadlocks-Characterization-Handling Deadlocks-Deadlock Prevention-Avoidance-Detection-Deadlock Recovery.

4. MEMORY MANAGEMENT **10**

Storage Hierarchy-Storage Management Strategies-Contiguous-Non Contigous Storage Allocation-Single User-Fixed Partition-Variable Partition-Swapping-Virtual Memory-Basic Concepts-Multilevel Organization-Block Mapping-Paging-Segmentation-Page Replacement Methods-Locality-Working Sets.

5. I/O AND FILE SYSTEMS **10**

Disk Scheduling-File Concepts-File System Structure-Access Methods-Directory Structure-Protection-Directory Implementation-Allocation Methods-Free Space Management-Case Study: Linux System.

Total: 45

TEXT BOOK :

1. Silberschatz and Galvin, Operating System Concepts, 6th Edition, John Wiley & Sons, Inc., 2004

REFERENCES :

1. Milankovic M., Operating System Concepts and Design, 2nd Edition, McGraw Hill, 1992
2. P.C.Bhatt, An Introduction to Operating Systems-Concepts and Practice, Prentice Hall Of India, 2004
3. H.M.Deitel, An Introduction to Operating Systems, 2nd Edition, Pearson Education, 2002

CA610 DESIGN AND ANALYSIS OF ALGORITHMS

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is to

- Introduce basic concepts of algorithms
- Introduce mathematical aspects and analysis of algorithms
- Introduce sorting and searching algorithms
- Introduce various algorithmic techniques
- Introduce algorithm design methods
- Teach how to apply the algorithms and design techniques to solve problems;
- Devise correct and efficient algorithms for solving a given problem
- Validate/verify algorithms
- Learn whether efficient algorithm exists for solving a given problem
- Write clear algorithms.
- Explain NP-Completeness and deal with NP-complete problems

Outcomes of the Course:

At the completion of this course, students will be able to

- Understand, explain, model and analyze a given software problem as a solution.
- Investigate whether the given algorithm found is most efficient.
- Formulate the time order analysis for an algorithm.
- Prove the correctness of an algorithm.
- Analyze the complexity of a given algorithm.
- Apply classical sorting, searching, optimization and graph algorithms.
- Compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem
- Understand various design methods and analyzing techniques will be learnt by the students
- Understand NP-Completeness and deal with NP-complete problems

1. INTRODUCTION **8**

Fundamentals of algorithmic problem solving – Important problem types – Fundamentals of the analysis of algorithm efficiency – analysis frame work – Asymptotic notations – Mathematical analysis for recursive and non-recursive algorithms.

2. DIVIDE AND CONQUER METHOD AND GREEDY METHOD **7**

Divide and conquer methodology – Merge sort – Quick sort – Binary search – Binary tree traversal – Multiplication of large integers – Strassen's matrix multiplication – Greedy method – Prim's algorithm – Kruskal's algorithm – Dijkstra's algorithm.

3. DYNAMIC PROGRAMMING **10**

Computing a binomial coefficient – Warshall's and Floyd' algorithm – Optimal binary search tree – Knapsack problem – Memory functions.

4. BACKTRACKING AND BRANCH AND BOUND **10**

Backtracking – N-Queens problem – Hamiltonian circuit problem – Subset sum problem – Branch and bound – Assignment problem – Knapsack problem – Traveling salesman problem.

5. NP-HARD AND NP-COMPLETE PROBLEMS **10**

P & NP problems – NP-complete problems – Approximation algorithms for NP-hard problems – Traveling salesman problem – Knapsack problem.

Total: 45

TEXT BOOK:

1. Anany Levitin "Introduction to the Design and Analysis of Algorithms" Pearson Education 2003.

REFERENCE:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L.Rivest, "Introduction to algorithms" Prentice Hall 1990.

CA611 OBJECT ORIENTED PROGRAMMING USING C++ **L T P C**
3 0 0 3

Objective of the Course:

The aim of the course is to

- Identify and practice the object-oriented programming concepts and techniques
- Practice the use of C++ classes and class libraries
- Get a clear understanding of object-oriented concepts.
- Understand object oriented programming through C++.
- Gain the basic knowledge on Object Oriented concepts.
- Develop applications using Object Oriented Programming Concepts.
- Implement features of object oriented programming to solve real world problems.

Outcomes of the Course:

At the completion of this course, students will be able to

- Understand object-oriented programming features in C++
- Apply these features to program design and implementation
- Understand object-oriented concepts and how they are supported by C++
- Gain some practical experience of C++
- Understand implementation issues related to object-oriented techniques
- Build good quality software using object-oriented techniques
- Understand the role of patterns in object-oriented design.
- Understand and solve logical & mathematical problems through C++ language. Strengthen knowledge of a procedural programming language. Design and develop solutions to intermediate level problems using the C language.
- Further develop skills in software development using a procedural language.

1. OOP PARADIGAM **8**

Programming Paradigms-Procedural Programming-Modularity-Exception Handling-Data Abstraction-User Defined Types-Concrete Types-Abstract

Types-Virtual Functions-Object Oriented Programming-Generic Programming-Containers-Algorithms.

2. INTRODUCTION TO C++ 11

Overview of C++-Classes and Objects-Friend Functions-Friend Classes-Inline Function-Static Members-Arrays-Pointers-References-Dynamic Allocation

3. OVERLOADING 7

Function Overloading-Overloading Constructor Functions-Copy Constructors-Default Argument-Operator Overloading-Member Operator Overloading-Overloading new and delete

4. ADDITIONAL FEATURES 10

Inheritance-Base Class-Access Control-Virtual Functions-Pure Virtual Functions-Templates-Generic Functions-Applying Generic Functions-Generic Classes-Exception Handling-C++ I/O Streams-File I/O-STL-Overview-Container Classes-Lists-Maps-Algorithms Using Functions and Objects-String Class.

5. DESIGN CONCEPTS 9

Role of Classes-Kinds of Classes-Concrete Types-Abstract Types-Nodes-Changing Interfaces-Object I/O-Actions-Interface Classes-Handles-Use Counts Applications frame works

Total: 45

TEXT BOOKS :

1. Herbert Schildt,"C++ The Complete Reference", Tata McGrawHill Edition, 2003 (unit 2, 3, 4)

2. Bjanne Stroustrup,"The C++ Programming Language",3rd Edition, Addison Wesley, 2000 (Unit 1 & 5)

REFERENCES

1. Robert Lafore."Waite Groups OOP in Turbo C++",Galgotia Publications, 2001

2. Stanley, B.Lippman,Jove Lagrie,"C++Primer",3rd Edition, Addison Wesley,1998

CA612 OBJECT ORIENTED PROGRAMMING LAB

L	T	P	C
0	0	6	2

Objective of the Course:

The aim of the course is to

- Understand and solve logical & mathematical problems through C++ language
- Strengthen knowledge of a procedural programming language.
- Design and develop solutions to intermediate level problems
- Develop their skills in software development using a procedural language
- Get programming in skill the object oriented technology with the usage of C++.

Outcomes of the Course:

On completion of this course the students will be able to

- Describe the syntax and semantics of the C++ programming language
- Explain the use of class and object.
- Eork in a team to analyze engineering problems and develop C++ programs for solving these problems.
- Use the basic utilities and facilities for software development.
- Debug and test software
- Develop a minor software in C++ language

1. Programs using Constructor and Destructor.
2. Creation of classes and use of different types of functions.
3. Count the number of objects created for a class using static member function.
4. Write programs using function overloading and operator overloading.
5. Programs using inheritance.
6. Program using friend functions.
7. Program using virtual function.
8. Write a program using exception handling mechanism.
9. Programs using files.
10. Programs using function templates.

CA613 ALGORITHMS AND DATA STRUCTURES LAB

L	T	P	C
0	0	6	2

Objective of the Course:

The aim of the course is to

- Introduce the concept of efficiency of an algorithm
- Study run-time efficiency of an algorithm
- Introduce Big-O notation
- Determine the Big-O of an algorithm

Outcomes of the Course:

On completion of this course the students will be able to:

- Analyze the complexity of a given algorithm.
- Apply classical sorting, searching, optimization and graph algorithms.
- Compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem.
- Develop program that implements krushkals algorithm, Binary Search, all types of sorting, prims algorithm, greedy algorithm, optimal binary search tree, Dijkstra's Algorithm, solution for n-queens problem, and program to implement traveling salesperson problem using dynamic programming.

ALGORITHMS:

1. Apply the divide and Conquer technique to arrange a set of numbers using merge sort method.
2. Perform Strassen's matrix multiplication using divide and conquer method.
3. Solve the knapsack problem using greedy method.
4. Construct a minimum spanning tree using greedy method.
5. Construct optimal binary search trees using dynamic programming method of problem solving.
6. Find the solution for traveling salesperson problem using dynamic programming approach.
7. Perform graph traversals.
8. Implement the 8 Queens Problem using backtracking.

9. Implement knapsack problem using backtracking.
10. Find the solution of traveling salesperson problem using branch and bound technique.

DATA STRUCTURES:

1. Represent the given sparse matrix using one dimensional array and linked list.
2. Create a Stack and do the following operations using arrays and linked lists

   (i) Push (ii) Pop (iii) Peep
3. Create a Queue and do the following operations using arrays and linked lists

   (i) Add (ii) Remove
4. Implement the operations on singly linked list, doubly linked list and circular linked list.
5. Create a binary search tree and do the following traversals

   (i) In-order (ii) Pre order (iii) Post order
6. Implement the following operations on a binary search tree.

   (i) Insert a node (ii) Delete a node
7. Sort the given list of numbers using heap and quick sort.
8. Perform the following operations in a given graph

   (i) Depth first search (ii) Breadth first search
9. Find the shortest path in a given graph using Dijkstra algorithm

EN611	COMMUNICATION SKILLS LABORATORY	L	T	P	C
		0	0	2	1

Objective of the Course:

The aim of the course is to

- Enable the students to speak English with correct accent and pronunciation.
- Interact effectively in real life situations and in workplace.
- Develop the writing ability of students by providing them the required practice.
- Improve the written communication skill so as to write reports, letters etc.

Outcomes of the Course:

This lab course will enable students to use 'good' English and perform the following

- Gather ideas and information, to organise ideas relevantly and coherently.
- Engage in debates.
- Participate in group discussions.
- Face interviews.
- Write project/research reports/technical reports.
- Make oral presentations.
- Write formal letters.
- Transfer information from non-verbal to verbal texts and vice versa.
- Take part in social and professional communication.

UNIT I LANGUAGE FUNDAMENTALS **3**

Tenses, Subject – Verb Agreement, Correction of Errors.

UNIT II ORAL COMMUNICATION **10**

Oral practice – Introducing oneself, Conversations, Role-play - Activities based on real life situations and professional situations such as marketing, advertising, etc. Debating on a topic, Group Discussion, Oral Presentation, Non-verbal communication, Mock Interviews, Conducting meetings, Participating in meetings- Phonetics- Correct Pronunciation.

UNIT III WRITTEN COMMUNICATION **6**

Writing a letter of application with resume- practical training- calling for quotations – placing an order – letter of complaint, Memoranda, Writing an email, Minutes - Report Writing - Project report - Writing a proposal.

UNIT IV LANGUAGE LABORATORY **6**

Language fundamental practices - Listening Comprehension, Reading Comprehension, Listening to correct pronunciation, Accent, Viewing models of Presentations, Interviews, Group Discussions in the language lab and practice in the class room.

Total : 25

REFERENCES :

1. A.J.Thomson & A.V. Martinet, “A Practical English Grammar” Oxford University Press, 1999.
2. Andrea J. Rutherford, “Basic Communication Skills for Technology”, second edition, Pearson Education, 2007.
3. P.K.Dutt, G. Rajeevan and C. L. N. Prakash, “ A Course in Communication Skills”, Cambridge University Press, India 2007.
4. Krishna Mohan and Meera Banerjee, “Developing Communication Skills “ Macmillan India Ltd. (reprinted 1994-2007.)
5. Riordan, Pauley, “Report Writing Today” , AIT B.S. Publisher, New Delhi(2000)
6. Gerson, Sharon, Steve m. Gerson, “Technical Writing: Process and Product”, Pearson Education, New Delhi(2004).
7. R.K. Bansal, J.P. Harrison, “Spoken English”, Orient Longman , Mumbai(1999)
8. Grant Taylor, “English Conversation Practice”Tata MCGraw Hill, New Delhi(1997).

SEMESTER – III

CA701	**INTERNET AND JAVA PROGRAMMING**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

Objective of the Course:

The aims of this course are to

- Learn basic internet concepts, fighting against spam, conferencing on the internet, planning and creating website.
- Learn the fundamentals of the capabilities of Java and to introduce encapsulation, polymorphism, and the Java language mechanism (classes and objects) to implement it.
- Make a study on graphics programming.
- Learn about event handling and exception handlings.
- Develop Java computer programs that performs various problem-solving algorithms.
- Apply the theory and professional knowledge of Java Programming to practice during hands-on laboratories.
- Develop the programming skills to use the Java object oriented programming methodology to produce quality computer based solutions to real problems.
- Work with collection of API and develop fast programs.
- Develop good multithreaded programs.

Outcomes of the Course:

On completion of this course the students will be able to

- Develop Java computer programs that perform various problem-solving algorithms.
- Improve the programming skills in Object Oriented language.
- Graphics programming and Internet Programming skills are developed.

1. BASIC INTERNET CONCEPTS **8**

Connecting to the Internet – Domain Name System - Exchanging E-mail – Sending and Receiving Files - Fighting Spam, Sorting Mail and avoiding e-mail viruses – Chatting and Conferencing on the Internet – Online Chatting - Messaging – Usenet Newsgroup – Internet Relay chat (IRC) – Instant Messaging - Voice and Video Conferencing.

2. WORLD WIDE WEB **8**

Overview – Web Security, Privacy, and site-blocking – Audio and Video on the web – Creating and Maintaining the Web – Web site creation concepts – Web Page Editors – Optimizing Web Graphics – Web Audio Files – Forms, Interactivity, and Database-Driven Web sites – File Transfer and downloading – FTP – Peer to Peer – Downloading and Installing software.

3. JAVA FUNDAMENTALS **8**

Java features – Java Platform – Java Fundamentals – Expressions, Operators, and Control Structures – Classes, Packages and Interfaces – Exception Handling.

4. PACKAGES **12**

AWT package – Layouts – Containers – Event Package – Event Model – Painting – Garbage Collection - Multithreading – Language Packages.

5. ADVANCED JAVA PROGRAMMING **9**

Utility Packages – Input Output Packages – Inner Classes – Java Database Connectivity - Servlets - RMI – Java Beans.

Total : 45

TEXT BOOKS :

1. Margaret Levine Young, “Internet and WWW”, 2nd Edition, Tata McGraw Hill, 2002. (Unit 1 & 2)

2. Herbert Schildt, The Complete Reference – Java 2 , 4th Edition, Tata McGraw Hill, 2001. (Unit 3, 4 & 5)

REFERENCES :

1. Keyur shah, “Gateway to Java Programmer Sun Certification”, Tata Mc Graw Hill 2002.

2. Deitel & Deitel, Java How to Program, Prentice Hall 1999.

CA702 COMPUTER GRAPHICS AND MULTIMEDIA SYSTEMS

L	T	P	C
3	0	0	3

Objective of the Course:

The goal of this course is to provide complete guidelines for theoretical and practical usage of computer graphics and multimedia, to provide students with an overview of the key concepts of digital production of animation and visual effects with reference to workflow, people and technology. Real time design can be performed for a several problems effectively. It will also enable students to be practically sound in area of 2D, 3D and animation.

Outcomes of the Course:

On completion of this course,

- The students will attain the complete knowledge in graphics & multimedia domain.
- Every student will understand the procedure to perform a design for real-time problems using various tools.
- Visualization, business graphics and effective multimedia aspects are also defined and understood by the students.
- Students can show their proficiency while working with Graphics and multimedia software's and tools.

1. INTRODUCTION **9**

Overview of Graphics System - Bresenham technique – Line Drawing and Circle Drawing Algorithms - DDA - Line Clipping - Text Clipping.

2. 2D TRANSFORMATIONS **9**

Two dimensional transformations – Scaling and Rotations - Interactive Input methods - Polygons - Splines – Bezier Curves - Window view port mapping transformation.

3. 3D TRANSFORMATIONS **9**

3D Concepts - Projections – Parallel Projection - Perspective Projection – Visible Surface Detection Methods - Visualization and polygon rendering – Color models – XYZ-RGB-YIQ-CMY-HSV Models - animation – Key Frame systems - General animation functions - morphing.

4. OVERVIEW OF MULTIMEDIA **9**

Multimedia hardware & software - Components of multimedia – Text, Image – Graphics – Audio – Video – Animation – Authoring.

5. MULTIMEDIA SYSTEMS AND APPLICATIONS **9**

Multimedia communication systems – Data base systems – Synchronization Issues – Presentation requirements – Applications – Video conferencing – Virtual reality – Interactive video – video on demand

Total : 45

TEXT BOOKS :

1. Hearn D and Baker M.P, "Computer graphics – C Version", 2nd Edition, Pearson Education, 2004(unit 1, 2 &3)

2. Ralf Steinmetz, Klara steinmetz, "Multimedia Computing, Communications and Applications", Pearson education, 2004(unit 4 & 5)

REFERENCES :

1. Siamon J. Gibbs and Dionysios C. Tsichritzis, "Multimedia programming", Addison Wesley, 1995.

2. John Villamil, Casanova and Leony Fernanadez, Eliar, "Multimedia Graphics", PHI, 1998.

CA703 MICROPROCESSORS AND APPLICATIONS

L	T	P	C
3	0	0	3

Objective of the Course:

To introduce features and technology of microprocessor based systems, gain assembly language programming, interfacing of memory and peripheral devices and gain knowledge about the architecture, instruction set, programming, addressing mode interfacing and applications of INTEL 8085, INTEL 8086 and INTEL 80386.

Outcomes of the Course:

Upon completion of this course the student shall be able to understand and do the following

- Draw a detailed architecture diagram of the INTEL 8085, INTEL 8086 and INTEL 80386.
- Develop a program in assembly language for the INTEL 8085 and INTEL 8086.
- Use all addressing modes in assembly language.
- Design interfacing logic to connect external devices to microprocessor.
- Design and develop a microprocessor based system for specific applications.

1. INTRODUCTION TO 8085 MICRO PROCESSOR 9

Evolution of the Microprocessor - INTEL 8085- Introduction- Register Architecture - Memory Addressing - 8085 Addressing Modes -8085 Instruction Set -Timing Methods 8085 Pins and Signals -8085 Instruction Timing and Execution –Interrupts-DMA- Serial port-8085 Based System Design.

2. INTRODUCTION TO 8086 MICROPROCESSOR 9

Introduction -8086 Architecture -8086 Addressing Modes -8086 Instruction Set –Data Movement Instructions Arithmetic and Logic Instructions - Program Control Instructions

3. 8086 MICROPROCESSOR INTERFACING 9

System Design Using 8086- Basic System concepts-Bus Cycle - Address and data bus concepts- interfacing with memories-RAM - EPROM - DRAMs - Programmed I/O : 8086-Based Microcomputer.

4. 80386 AND PENTIUM MICRO PROCESSORS 9

Introduction to Intel 80386- Basic Programming model - Memory Organisation - I/O Space - 80386 pins and signals- Bus transfer techniques - 80386 Modes – Introduction to Intel Pentium Microprocessor: Block diagram and Registers.

5. PERIPHERAL INTERFACING 9

Keyboard Display Interface-Hex key and display interface to 8085, 8279 Keyboard display controller chip- Printer Interface: LR 7040 Printer interface using 8295 printer controller-CRT controller interface: CRT Fundamentals, 8275 CRT Controller- Coprocessors.

Total : 45

TEXT BOOK :

1. Mohamed Rafiquzzaman “Introduction to Microprocessors and Microcomputer- Based System Design” 2nd edition, CRC Press,1995.

REFERENCES :

1. Walter A.Triebel, Avtar Singh, “The 8088and8086 Microprocessors Programming, Interfacing, Software, Hardware and Applications”, Prentice Hall of India Pvt. Ltd., 2002.

2. Barry B.Brey,”The INTEL microprocessors 8086/8088, 80186, 80286, 80386 and 80486 Architecture, Programming and Interfacing,” Prentice Hall of India, 2001.

CA704 DISTRIBUTED COMPUTING **L T P C**
3 0 0 3

Objective of the Course:

To introduce new geographic information technologies distributed, ubiquitous, and mobile allowing geographic information to be accessed, analyzed, and used in decision-making anywhere, and any time.

Outcomes of the Course:

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

- Understand and apply theoretical foundations of programming concurrent distributed computing systems,
- Compare communication and synchronization aspects in actor-, process-, and object-oriented concurrent programming languages,
- Design and develop the computing systems for distributed execution over Internet-connected run-time environments,
- Critique research papers on distributed computing over the Internet and identify open research problems.

1. INTRODUCTION **9**

Characterization of Distributed Systems - Examples - Resource Sharing and the Web - Challenges - System Models - Architectural and Fundamental Models - Networking and Internetworking - Types of Networks - Network Principles - Internet Protocols - Case Studies.

2. PROCESSES AND DISTRIBUTED OBJECTS **9**

Interprocess Communication - The API for the Internet Protocols - External Data Representation and Marshalling - Client-Server Communication - Group Communication - Case Study - Distributed Objects and Remote Invocation - Communication Between Distributed Objects - Remote Procedure Call - Events and Notifications - Java RMI - Case Study.

3. OPERATING SYSTEM ISSUES – I **9**

The OS Layer - Protection - Processes and Threads - Communication and Invocation – OS Architecture - Security - Overview - Cryptographic Algorithms - Digital Signatures - Cryptography Pragmatics - Case Studies - Distributed File Systems - File Service Architecture - Sun Network File System - The Andrew File System

4. OPERATING SYSTEM ISSUES – II **9**

Name Services -Domain Name System - Directory and Discovery Services - Global Name Service - X.500 Directory Service - Clocks, Events and Process States - Synchronizing Physical Clocks - Logical Time And Logical Clocks - Global States - Distributed Debugging - Distributed Mutual Exclusion – Elections – Multicast Communication Related Problems.

5. DISTRIBUTED TRANSACTION PROCESSING **9**

Transactions - Nested Transactions - Locks - Optimistic Concurrency Control - Timestamp Ordering - Comparison - Flat and Nested Distributed Transactions - Atomic Commit Protocols - Concurrency Control in Distributed Transactions - Distributed Deadlocks - Transaction Recovery - Overview of Replication And Distributed Multimedia Systems

Total: 45

TEXT BOOK:

1. George Coulouris, Jean Dollimore and Tim Kindberg, Distributed Systems Concepts and Design, Pearson Education, 3rd Edition, 2002.

REFERENCES:

1. Sape Mullender, Distributed Systems, Addison Wesley, 2nd Edition, 1993.
2. Albert Fleishman, Distributes Systems- Software Design and Implementation, Springer-Verlag, 1994
3. M.L.Liu, Distributed Computing Principles and Applications, Pearson Education, 2004.
4. Andrew S Tanenbaum , Maartenvan Steen,Distibuted Systems –Principles and Pardigms,Pearson Education, 2002
5 Mugesh Singhal, Niranjan G Shivaratri, Advanced Concepts in Operating Systems,Tata McGraw Hill Edition, 2001.

CA705	JAVA PROGRAMMING AND GRAPHICS & MULTIMEDIA LAB	L	T	P	C
		0	0	6	2

Objective of the Course:

The aims of the course are

- To develop Java computer programs that performs various problem-solving algorithms.
- To apply the theory and professional knowledge of Java Programming to practice during hands-on laboratories.
- To develop the programming skills using the object oriented programming methodology to produce quality computer based solutions to real problems.
- To utilize the advance features of Java technology.
- To Work with collection API and develop fast programs.
- To develop good multithreaded programs

Outcomes of the Course:

On completion of this course,

- The programming skill of students in object oriented language especially in java gets improved and they become a good internet programmer.
- They will be able to utilize the advance features of Java technology.
- They will be able to work with collection of API and develop fast programs.
- They will be able to develop good multithreaded programs

JAVA PROGRAMMING:

1. Program to illustrate the use of overloading and overriding.
2. Program to implement the concept of inheritance
3. Program to illustrate the use of multi threading
4. Program to implement the concept of Interfaces and packages.
5. Generate the program using exceptions handling mechanism..
6. Implement the file operations.
7. Program using Applets.
8. Program using JDBC.

GRAPHICS AND MULTIMEDIA:

1. Write a Java program for Line drawing using Bresenham, DDA Line Drawing Algorithms.
2. Write a Java program for Circle Drawing using Bresenham Circle Drawing Algorithms.
3. Write a Java program for Line Clipping using Cohen-Sutherland Line clipping algorithm.
4. Write a Java program for 2D and 3D Transformations like Translations and Scaling and Rotations.
5. Create Frame by Frame Animations using multimedia authoring tools.
6. Develop a presentation for a product using techniques like Guide Layer, masking and onion Skin using authoring tools.
7. Create a JPEG image which demonstrates the various features of an image editing tool.

CA706	MICROPROCESSORS LAB	L	T	P	C
		0	0	6	2

Objective of the Course:

This course has been designed to provide students with enough knowledge in the fundamentals of Microprocessors, to learn programming in assembly language and to use Interfaces with Microprocessor.

Outcomes of the Course:

On completion of the course, students will be able to

- Attain knowledge in Microprocessors
- Program in assembly language efficiently
- Make effective use of microprocessors in various applications
- Utilize various Interfaces along with Microprocessor

1. Write an assembly language program to perform arithmetic operations on block of data using Hexadecimal numbers.
2. Write an assembly language program to perform arithmetic operations on block of data using BCD numbers.
3. Write an assembly language program to perform byte and string manipulation.
4. Write an assembly language program to interface Programmable Peripheral Interface.
5. Write an assembly language program to interface Programmable Timer.
6. Write an assembly language program to interface Programmable Communication Interface.
7. Write an assembly language program to interface Keyboard/Display Controller.
8. Write a program to Perform Power on Self Test.
9. Write a program for floppy disk trouble shooting.
10. Write a program for printer trouble shooting.

CA713	SOFT SKILLS	L	T	P	C
		0	0	2	1

Objective of the Course:

This course is intended to provide

- Attitude Control and Quick Personal Self Esteem Improvement
- Interview Motivation and Developing Personal Confidence using NLP
- Effective Answering and Maintaining Fluent Communication
- Positive Body Language
- Effective Resume Creation
- Leadership Skills, Team Player Skills and Career Plan to HR

Outcomes of the Course:

By the end of this course students will be able to develop the following:

- Motivation
- Awareness of Real World Industry and Situations
- Conscious about Self-Awareness
- Practical Visualizations of peer group relationship
- Neuro-Linguistic Programming Basics
- Developing Self Esteem and Self Motivation
- Confident Goal Setting
- Positive Attitude Development and Positive Thinking
- Developing Inner Achievement Mindset
- Leadership skills
- Types of Leadership
- Leadership Process and Thinking
- Innovative Thinking
- Role of Competency, Discipline, Planning and Ethics
- Creating Cooperation and Trust in Team Building Process
- Mentoring
- Effective Communication

- Eliminating Stage Fright
- Increasing Fluency
- Increasing Focus while Listening
- How to communicate as a Follower
- How to communicate as a Leader
- Assertive and Polite Communication
- Fluency with increased vocabulary
- Group Discussion Etiquette
- Advanced Group Discussions
- Giving a positive Body Language
- Interviews: Clear Speaking
- Interviews: Handling Pressure

UNIT 1 : MOTIVATION **5**

Awareness of Real World Industry and Situations - Conscious Self-Awareness - Practical Visualizations - Neuro- Linguistic Programming Basics - Developing Self Esteem and Self Motivation - Confident Goal Setting - Positive Attitude Development and Positive Thinking - Developing Inner Achievement Mindset.

UNIT 2 : LEADERSHIP SKILLS **4**

Types of Leadership - Leadership Process and Thinking - Innovative Thinking - Role of Competency, Discipline, Planning and Ethics - Creating Cooperation and Trust in Team Building Process – Mentoring.

UNIT 3 : EFFECTIVE COMMUNICATION I **4**

Eliminating Stage Fright - Increasing Fluency - Increasing Focus while Listening - How to communicate as a Follower - How to communicate as a Leader - Assertive and Polite Communication.

UNIT 4 : EFFECTIVE COMMUNICATION II **5**

Fluency with increased vocabulary - Group Discussion Etiquette - Advanced Group Discussions - Giving a positive Body Language - Interviews: Clear Speaking - Interviews: Handling Pressure.

Total : 18

REFERENCES :

1. Charles Faulkner and Steve Andreas, "NLP: The New Technology of Achievement", Harper Paperbacks publishers, 1996.
2. Sarvesh Gulati, "Corporate Soft Skills", Rupa and Co publishers.
3. P.K.Dutt, G. Rajeevan and C. L. N. Prakash, “A Course in Communication Skills”, Cambridge University Press, India 2007.

SEMESTER – IV

		L	T	P	C
CA707	**OBJECT ORIENTED ANALYSIS AND DESIGN**	**3**	**0**	**0**	**3**

Objective of the Course:

The objective of this course is to enhance correct way to do object-orientation. It also explores the basic code qualities of cohesion, coupling, redundancy, testability, readability, encapsulation and dispersion. While not a course on patterns, object-orientation is taught in a way consistent with how design patterns suggest identifying and defining objects.

Outcomes of the Course:

By the end of this course students will be able to

- Apply an iterative process such as the Unified Process.
- Analyze software requirements and document those using Use Cases.
- Perform software analysis and record the results using UML notation.
- Perform software design and record the results using UML notation.
- Apply object-oriented patterns.
- Discuss how object-oriented software development affects testing and quality.

1. INTRODUCTION **9**

An overview – Object basics – Object state and properties – Behavior – Methods – Messages – Information hiding – Class hierarchy – Relationships – Associations – Aggregations- Identity – Dynamic binding – Persistence – Metaclasses – Object oriented system development life cycle.

2. METHODOLOGY AND UML **9**

Introduction – Survey – Rumbugh, Booch, Jacobson methods – Patterns – Frameworks – Unified approach – Unified modeling language – Static and Dynamic models – UML diagrams – Class diagram – Usecase diagrams – Dynamic modeling – Model organization – Extensibility.

3. OBJECT ORIENTED ANALYSIS **9**

Identifying Usecase – Business object analysis – Usecase driven object oriented analysis – Usecase model – Documentation – Classification – Identifying object, relationships, attributes, methods – Super-sub class – A part of relationships Identifying attributes and methods – Object responsibility

4. OBJECT ORIENTED DESIGN **9**

Design process – Axions – Cololllaries – Designing classes – Class visibility – Refining attributes – Methods and protocols – Object storage and object interoperability – Databases – Object relational systems – Designing interface objects – Macro and Micro level processes – The purpose of a view layer interface.

5. SOFTWARE QUALITY **9**

Quality assurance – Testing strategies – Object orientation testing – Test cases – Test Plan – Debugging principles – Usability – Satisfaction – Usability testing – Satisfaction testing.

Total : 45

TEXT BOOK :

1. Ali Bahrami, “Object Oriented System Development”, McGraw Hill International Edition, 1999.

REFERENCES :

1. Craig Larman, Applying UML and Patterns, 2nd Edition, Pearson, 2002.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language User Guide”, Addison Wesley Long man, 1999.
3. Bernd Bruegge, Allen H. Dutoit, Object Oriented Software Engineering using UML, Patterns and Java, Pearson 2004

CA708	WEB GRAPHICS	L	T	P	C
		3	0	0	3

Objective of the Course:

This course is to

- Understand the importance of the web as a medium of communication.
- Understand the principles of creating an effective web page
- Learn the language of the web: HTML.
- Develop skills in using WYSIWYG web development software (cascading style sheets)
- Develop skills in digital imaging (Adobe Photoshop)
- Develop skills in animation (Flash)

Outcomes of the Course:

Once the course is completed the students will be able to

- Work effectively and flexibly with Photoshop and Flash tools to create stylish web graphics
- Create and optimize images for the web.
- Use filters and layer effects both as tools and effects
- Export from Photoshop using the most appropriate file formats for the web.

1. INTRODUCTION **9**

HTML coding - Basic web graphics - Web page design and site building - Image maps - Adding multimedia to the web.

2. PAINT SHOP PRO/PHOTOSHOP **9**

Introduction - Image Basics - File Formats - GIF - JPEG - Color Palette - Layers - Creating new Images - Brushes - Grids - Scaling Images - Moving and Merging Layers - Tool Palette - Screen capturing - Grey styling - Using style Palette - Animation.

3. IMAGE HANDLING **9**

Scanning Images - Adding Text to the images - Designing icons - Creating background images - Color models - Color depths - Color calibration - Creating gradients - Oil paint effect.

4. MULTIMEDIA **9**

Creating clippings - Animations with sound effects - Adding audio or Video - Windows Media Player ActiveX Control - Agent control - Embedding VRML in a web page - Real Player ActiveX control.

5. APPLICATIONS **9**

Creating web site with a particular theme using all the utilities - Graphics - Animations and Interaction.

Total: 45

TEXT BOOKS :

1. Richard Schrand, Photoshop 6 Visual Jumpstrat, Adobe Press 2000. (Unit 1,2 & 3)

2. James L. Mohles, Flash 5.0 Graphics, Animation & Interaction, Macromedia 2000. (Unit 4 & 5)

REFERENCES :

1. Internet and World Wide Web How to program , Deitel – Prentice Hall 2003

2. Robert Reinhardt, Jon Warren Lentz ,"Flash 5 Bible", Hungry Minds Inc, 2001.

CA709 UNIX AND NETWORK PROGRAMMING

L	T	P	C
3	0	0	3

Objective of the Course:

The main objective of this course is to learn and get familiar with the UNIX operating system and UNIX process environment. The creation, communication and execution of process are studied with well defined examples through this course. Understanding of the term socket and socket programming is pursued efficiently.

Outcomes of the Course:

On completion of this course,

- Basic UNIX operating system concepts and file conventions can be understood by the students.
- The students will attain the complete knowledge in network communication in UNIX platform.
- Every student will learn about UNIX software such as red-hat Linux and Solaris.
- Different types of networking and their functionality will be understood by the students.
- Students get the broad knowledge on network details of a system which is configured as UNIX supported components.

1. INTRODUCTION & FILE SYSTEM **9**

Overview of UNIX OS - File I/O – File Descriptors – File sharing - Files and directories – File types - File access permissions – File systems – Symbolic links - Standard I/O library – Streams and file objects – Buffering - System data files and information - Password file – Group file – Login accounting – system identification.

2. PROCESSES **9**

Environment of a UNIX process – Process termination – command line arguments - Process control – Process identifiers - Process relationships terminal logins – Signals -threads.

3. INTERPROCESS COMMUNICATION **9**

Introduction - Message passing (SVR4)- pipes – FIFO – message queues - Synchronization (SVR4) – Mutexes – condition variables – read – write locks – file locking – record locking – semaphores –Shared memory(SVR4).

4. SOCKETS **9**

Introduction – transport layer – socket introduction - TCP sockets – UDP sockets - raw sockets – Socket options - I/O multiplexing - Name and address conversions.

5. APPLICATIONS **9**

Debugging techniques - TCP echo client server - UDP echo client server - Ping - Trace route - Client server applications like file transfer and chat.

Total: 45

TEXT BOOKS :

1. W.Richard Stevens, Advanced programming in the UNIX environment, Addison Wesley, 1999.(Unit 1,2 &3)

2. W. Stevens, Bill Fenner, Andrew Rudoff, “Unix Network Programming”, Volume 1,The Sockets Networking API,3rd Edition, Pearson education, Nov 2003.(unit 4 & 5)

REFERENCE BOOK :

1. Meeta Gandhi, Tilak Shetty and Rajiv Shah – The ‘C’ Odyssey Unix –The open Boundless C, 1st Edition, BPB Publications1992.

CA710	XML AND WEB SERVICES	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of this course is to

- Describe how web services are used and the technologies used to develop them.
- Describe the role of SOAP in building web services, the basics of the SOAP protocol, and the structure of a SOAP document.
- Explain the role of WSDL and UDDI in building web services and be familiar with their usage.
- Design a web service, taking different design considerations into account.
- Design a .NET web service, taking advanced design considerations into account.
- Define and address the various security aspects of building and accessing web services.
- Design and deploy .NET web services.
- Explain the use of UDDI registries and WSDL in creating web-services clients.
- Describe the data types, access methods, and protocols used when accessing web services.
- Design a web service using Microsoft .NET.
- Evaluate and analyze web-service parameters and make relevant design decisions about web-service clients.

Outcomes of the Course:

Students will learn:

- The role of web services in commercial applications
- The principles of web service provision
- Use of Java for implementing web services
- Use of BPEL (Business Process Execution Logic) and WSDL (Web Service Description Language) for implementing web services

- **Transferable skills in:**
- Understanding of Internet service provision
- Understanding and experience of designing and implementing distributed services

1. INTRODUCTION **9**

Role Of XML – XML and The Web – XML Language Basics – SOAP – Web Services – Revolutions Of XML – Service Oriented Architecture (SOA).

2. XML TECHNOLOGY **9**

XML – Name Spaces – Structuring With Schemas and DTD – Presentation Techniques – Transformation – XML Infrastructure.

3. SOAP **9**

Overview Of SOAP – HTTP – XML-RPC – SOAP: Protocol – Message Structure – Intermediaries – Actors – Design Patterns And Faults – SOAP With Attachments.

4. WEB SERVICES **9**

Overview – Architecture – Key Technologies - UDDI – WSDL – ebXML – SOAP And Web Services In E-Com – Overview Of .NET And J2EE.

5. XML SECURITY **9**

Security Overview – Canonicalization – XML Security Framework – XML Encryption – XML Digital Signature – XKMS Structure – Guidelines For Signing XML Documents – XML In Practice.

Total: 45

TEXT BOOK:

1. Frank. P. Coyle, XML, Web Services And The Data Revolution, Pearson Education, 2002.

REFERENCES:

1. Ramesh Nagappan, Robert Skoczylas and Rima Patel Sriganesh, “Developing Java Web Services”, Wiley Publishing Inc., 2004.
2. Sandeep Chatterjee, James Webber, “Developing Enterprise Web Services”, Pearson Education, 2004.
3. McGovern, et al., “Java Web Services Architecture”, Morgan Kaufmann Publishers, 2005.

CA711 UNIX AND NETWORK PROGRAMMING LAB

L	T	P	C
0	0	6	2

Objective of the Course:

The main objective of this course is to learn and get familiar with the Unix operating system through Red hat Linux. UNIX basic commands working and their purpose will be defined .The communication system across the network will be discussed and represented practically. The interprocess communication is implemented using pipe, named pipe, message queue, semaphore and shared memory concepts. The socket programming with TCP and UDP protocols are practically implemented.

Outcomes of the Course:

On completion of this course,

- Basic UNIX commands their syntax and purpose can be well understood by the students.
- The students will attain the hands on experience in working with Red hat Linux.
- The inter-process communication concepts were implemented practically by the students.
- Different types of networking and their functionality is understood by the students.
- Students get the practical knowledge on network details of a system which is configured as UNIX supported components.

1. Program using basic network commands
2. Program using system calls : create, open, read, write, close, stat, fstat, lseek
3. Program to implement inter process communication using pipes
4. Program to perform inter process communication using message queues
5. Program to perform inter process communication using shared memory
6. Program to perform synchronization using semaphores
7. Program to capture packets : sniffer
8. Program using TCP sockets (Client and Server)
9. Program using UDP sockets (Client and Server)
10. Program using URL class to download webpages

CA712	XML AND WEB SERVICES LAB	L	T	P	C
		0	0	6	2

Objective of the Course:

The aim of the course is

- To provide the knowledge necessary to build and validate XML.
- To with XML being a core technology in Web Services, this will give students a background in the underlying technologies.
- To cover the basics of XML, Schemas, WSDL, and Web Services concepts.

Outcomes of the Course:

On completion of this course, students will be able to

- Build effective XML documents
- Build DTD documents to validate XML
- Build Schema documents to validate XML
- Describe services using WSDL
- Build and consume Web services
- Understand the role of web services in commercial applications
- Understand the principles of web service provision
- Understand the use of Java for implementing web services

1. Create an XML document to store an address book.
2. Create an XML document to store information about books and create the DTD files.
3. Create an XML schema for the book's XML document from exercise 2.
4. Create an XML document to store resumes for a job web site and create the DTD file
5. Present the book's XML document using cascading style sheets (CSS).
6. Write an XSLT program to extract book titles, authors, publications, book rating from the book's XML document and use formatting.
7. Use Microsoft DOM to navigate and extract information from the book's XML document.

8. Use Microsoft DSO to connect HTML form or VB form to the book's XML document and display the information.

9. Create a web service for temperature conversion with appropriate client program.

10. Create a web service for currency conversion (at five currencies) with appropriate client program.

SEMESTER – V

CA801	PRINCIPLES OF COMPILER DESIGN	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is

- To introduce the major concept areas of language translation and compiler design
- To develop an awareness of the function and complexity of modern compilers
- To understand the phases of the compilation process and be able to describe the purpose and implementation approach of each phase.
- To give students practical exposure to aspects of theoretical Computer Science including Languages, Grammars, and Machines.
- To exercise and reinforce prior programming knowledge with a non-trivial programming project to construct a compiler.

Outcomes of the Course:

On completion of this course, students will be able to

- Demonstrate a working understanding of the process of scanning through the identification of the tokens of a programming language, the construction of regular expressions to define tokens, the construction of finite state automata to recognize tokens and the writing of a functioning scanner to automatically identify the tokens in a program
- Demonstrate a working understanding of the process of parsing through the application of EBNF to the definition of a programming language, the conversion of an EBNF definition of a programming language into an LL(1) context free grammar, the construction of an LL(1) table for an LL(1) grammar and the writing of a functioning recursive descent compiler based on said LL(1) grammar
- Demonstrate a working understanding of the process of semantic analysis through the construction of semantic records based on parse trees, the construction of symbol tables, the organization of run time memory and the writing of a semantic analyzer for a compiler.
- Design, analyze, implement and test a working compiler for a small language
- Develop a compiler with all the phases

1. INTRODUCTION **9**

Basic concepts - Grammar - Language - Parts of a compiler – Grouping of phases - Compiler construction tools.

2. LEXICAL ANALYZER **9**

Role of a lexical analyzer – Input buffering - Specification and recognition of tokens - Finite automata - Regular expression to finite automation – Optimization of DFA-based pattern matchers-Use of a tool for generating lexical analyzer.

3. SYNTAX ANALYZER **9**

Role of a parser - Context-free grammars - Top-down parsing - Bottom-up parsing - Use of a tool to generate parsers.

4. INTERMEDIATE CODE GENERATION **9**

Intermediate languages - Declaration - Assignment statements - Boolean expressions - Flow control statements –Back patching.

5. CODE GENERATION **9**

Introduction to optimization techniques - Issues in the design of a code generator - Run-time storage management - Design of a simple code generator.

Total:45

TEXT BOOK :

1. A.V. Aho, Ravi Sethi, J.D. Ullman, Compilers - Principles, Techniques and Tools, Addison- Wesley, 1988.

REFERENCES:

1. Fischer Leblanc, Crafting Compiler, Benjamin Cummings, Menlo Park, 1988.
2. Kennath C.Louden, Compiler Construction Principles and Practice, Vikas publishing House, 2003.
3. Allen I. Holub, Compiler Design in C, Prentice Hall of India, 2001.

MS084	ORGANIZATIONAL BEHAVIOUR	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is

- To enable the students to understand the need of Organizational Behaviour in technical environment
- To explain concepts, principles and techniques relating to different functional areas of Organizational Behaviour
- To make the students to understand the need for applying the concepts of Organizational Behaviour to improve the overall performance.

Outcomes of the Course:

Students would be encouraged to work in team and also to lead and come up with more innovative ideas.

1. LEADERSHIP **9**

Technical Leadership - Leader's Goal, Conviction, Vision - Transformational and Transactional Leadership - Leader's Vision - Professionalism : Importance, Elements - Managing Awareness - Performance - Manager's Role in Professionalism.

2. MANAGING TECHNICAL AND PROFESSIONAL PEOPLE **9**

Goals of Engineers and Scientists - Work Assignment - Need for Influence - Professional Career and Goals - Age and Creativity - Performance - Motivation - Employee Partnership - Career Risks - Technical Competence - Professional Discipline - Manager's Role in Professional Discipline - Guidelines.

3. IDENTIFICATION AND DEVELOPMENT OF TALENTED PEOPLE **9**

Talented Professionals – Importance - Characterization - Identification – Assessment and Recognizing Talent - Development - Development Needs - Counseling.

4. INNOVATION **9**

The Importance of Innovation - Risk of Failure - Nature of Creativity - Imagination - Managing Innovative Teams - Needs of Creative Teams - Team Dynamics - A Software Development Example - Manager's Responsibility - Team's Personal Needs - Political versus Technical Solutions - Team Synergism.

5. TEAM ENVIRONMENT AND RECOGNITION **9**

Innovative Team Environment -Award Programs - Recognition Programs - An Example Award Plan - Industry Award Plans - Award Guidelines - Incentive Plans - A Caution on Recognition Programs

Total: 45

TEXT BOOK :

1. Watts S. Humphrey, “Managing Technical People: Innovation, Teamwork, and the Software Process”, Addison-Wesley, 1996.

REFERENCES :

1. Saiyadain, M.S. Organizational Behaviour, 2009, Tata McGraw Hill.
2. Prasad C.M, Organizational Behaviour, 2002, Publisher: Sultan Chand and Sons.

CA803 MIDDLEWARE TECHNOLOGIES

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is to learn

- The fundamentals of three tier architecture
- About the Client/Server Architecture
- About the concepts of Enterprise Java Beans
- About the concepts of ASP.Net
- About the concepts of VB.Net

Outcomes of the Course:

On completion of the course, the students will understand

- What is Middleware and what are the different types of middleware.
- The basics of Client/Server Architecture.
- The role played by EJB in developing different types of applications.
- Importance of .Net Framework.
- The basics of ASP.
- The basics of VB.Net and ADO.Net

1. CLIENT / SERVER CONCEPTS (9)

Client – Server – File Server, Database server, Group server, Object server, Web server. Middleware – General middleware – Service specific middleware. Client / Server Building blocks – RPC – Messaging – Peer – to- Peer.

2. EJB ARCHITECTURE (9)

EJB – EJB Architecture – Overview of EJB software architecture – View of EJB – Conversation – Building and Deploying EJBs – Roles in EJB.

3. EJB APPLICATIONS (9)

EJB Session Beans – EJB entity beans – EJB clients – EJB Deployment – Building an application with EJB.

4. ASP.NET (9)

Basics of HTML - Introduction to Microsoft Visual Studio 2005 API - Overview of the .NET 2.0 Framework - Language Primer - Understanding Web Form Pages - Form Validation - Building Business Components - Building User Controls - Designing Web Sites with Master Pages - Adding Standard Navigation Elements - Securing ASP.NET 2.0 Web Sites - Building Applications.

5. VB.NET AND ADO.NET (9)

Overview of VB.Net - Variables, Constants, Operators and Calculations - Conditions, Switches and Loops - Arrays - Structures and Enumerators - Class, Methods, Objects and Namespaces - Method overloading and Object overloading - Delegates - Inheritance - Exception Handling - Multi-Threading - Overview of ADO.NET 2.0 - Displaying and Editing Data - Crystal Report - Building Applications.

Total No of periods: 45

TEXT BOOKS :

1. Robert Orfali, Dan Harkey and Jeri Edwards, "The Essential Client/Server Survival Guide", Galgotia Publications Pvt. Ltd., 2002. (Unit 1)
2. Tom Valesky,"Enterprise Java Beans",Pearson Education, 2002.(Unit 2 & 3)
3. DreamTech Software Team,"ASP.Net 2.0 Black Book", DreamTech Press, 2006. (Unit 4)
4. Jeffrey R.Shapiro,"Visual Basic.Net - Complete Refrence" (Unit 5)

REFERENCE:

1. Greg Buczek,"ASP.Net Developer Guide", Tata McGraw-Hill.

CA804	MIDDLEWARE LAB	L	T	P	C
		0	0	6	2

Objective of the Course:

The aim of the course is

- To explain the main principles of component technology,
- To explain how Object-Oriented middleware can be used for parallel and distributed programming,
- To state and detail how Object-Oriented middleware, together with components turn out to be very effective for the Grid.

Outcomes of the Course:

On completion of the course, students will have

- Ability to bring together resources across dissimilar networks or computing platforms.
- Ability to develop an independent programming interface without Application Programme Interface (API)

1. Create a distributed application to download various files from various servers using RMI
2. Create a Java Bean to draw various graphical shapes and display it using or without using BDK
3. Develop an Enterprise Java Bean for Banking operations
4. Develop an Enterprise Java Bean for Library operations
5. Create an Active-X control for File operations
6. Develop a middleware component for retrieving Stock Market Exchange information using .NET
7. Develop a middleware component for retrieving Weather Forecast information using .NET
8. Develop a component for converting the currency values using VB.NET
9. Develop a component for encryption and decryption using VB.NET
10. Develop a component for retrieving employee information from a Database Table (Oracle or Access) using VB.NET and ADO.NET

CA805	PROJECT PHASE – I	L	T	P	C
		0	0	6	*

SEMESTER VI

CA806	PROJECT PHASE - II	L	T	P	C
		0	0	36	24

Objective of the Course:

The objective of the project is to help the student to develop the ability to apply theoretical and practical tools / techniques to solve real life problems related to industry, academic institutions and research laboratories.

Outcomes of the Course:

After the completion of this project work, the student should be able to

- Describe the Systems Development Life Cycle (SDLC).
- Evaluate systems requirements.
- Complete a problem definition.
- Evaluate a problem definition.
- Determine how to collect information to determine requirements.
- Perform and evaluate feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and Operational feasibility for the project.
- Work on data collection methods for fact finding.
- Construct and evaluate data flow diagrams.
- Construct and evaluate data dictionaries.
- Evaluate methods of process description to include structured English, decision tables and decision trees.
- Evaluate alternative tools for the analysis process.
- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.
- Decide the S/W requirement specifications and H/W requirement specifications.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.
- Design and evaluate system outputs.

- Design and evaluate systems inputs.
- Design and evaluate validity checks for input data.
- Design and evaluate user interfaces for input.
- Design and evaluate file structures to include the use of indexes.
- Estimate storage requirements.
- Explain the various file update processes based on the standard file organizations.
- Decide various data structures.
- Construct and evaluate entity-relationship (ER) diagrams for RDBMS related projects.
- Perform normalization for the normalized tables for RDBMS related projects
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Identify documentation requirements and prepare and evaluate systems documentation.
- Perform various systems testing techniques/strategies to include the phases of testing.
- Perform systems implementation and identify its key problems.
- Generate various reports.
- Prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
- Decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.
- Work effectively as an individual or as a team member to produce correct, efficient, well-organized and documented programs in a reasonable time .
- Recognize problems that are amenable to computer solutions, and knowledge of the tools necessary for solving such problems.

- Develop of the ability to assess the implications of work performed.
- Get good exposure and command in one or more application areas and on the software
- Develop quality software using the software engineering principles

ELECTIVES

CAY001 MANAGEMENT INFORMATION SYSTEMS

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is

- To understand the basic principles of how the information system support the management in the various arena in the business units. MIS is the basic necessity of organization and encompasses all decisions in life, more particularly, business decisions and implementation.
- To provide an understanding of the Information Systems (IS) management framework of E-business.
- To focus on best practices, tools and models to implement an effective management system
- To provide insights on how to develop and implement enterprise-wide IT strategies, initiatives and programs
- To explore MIS subsystems and technologies including hardware, software and networking.

Outcomes of the Course:

At the end of this course, the students will be able to

- Understand the importance of determining information system requirements for all management levels by developing and understanding of the differences between various types of information systems
- Understand how information systems are developed
- Understand the computer revolution and its impact on the way business is conducted
- Become familiar with critical-thinking skills in identifying information systems problems and how to investigate existing literature about hardware and software solutions to problems.
- Know the components and functions of computer systems, both hardware and software.
- Become familiar with the advances in networking, data communications and the Internet and how they affect the way business is conducted.

- Identify which information technology tools are used to solve various business problems.

1. SYSTEM CONCEPTS **7**

Definition – Computer based user machine system – Integrated system – Need for a database – Utilization of models – Evolution – Subsystems – Organizational subsystems – Activities subsystems.

2. ORGANIZATIONAL STRUCTURE **9**

Basic model – Hierarchical – Specialization – Formalization – Centralization – Modifications of basic organizational structure – Project organization – Lateral relations – Matrix organization – Organizational culture and power organizational change

3. STRUCTURE OF MIS **10**

Operating elements – Physical components – Processing functions – Outputs – MIS support for decision making – Structured programmable decisions – Unstructured non-programmable decisions – MIS structure based on management activity and organizational functions – Synthesis of MIS structure

4. SYSTEM SUPPORT **10**

Data representation – Communication network – Distributed systems – Logical data concepts – Physical storage devices – File organizations – Data base organization – Transaction processing

5. DEVELOPMENT AND MANAGEMENT **9**

A contingency approach to choosing an application – Developing strategy – Lifecycle definition stage – Lifecycle development stage – Lifecycle installation and operation stage – Project management

Total: 45

TEXT BOOK :

1. Gordon B. Davis, Margrethe H. Olson, Management Information Systems: Conceptual foundations, Structure and development –2nd Edition – Tata-Mc Graw hill International book company, 2000

REFERENCES :

1. E.Wainright Martin, Carol V. Brown, Danial W. DeHayes, Jeffrey A. Hoffer, William C. Perkins, "Managing Information Technology" 3rd Edition, Prentice Hall International edition 1999.

2. Harold Koontz, Heinz Weihrich, "Essentials of Management", 5th Edition, Tata McGraw Hill 1998.

CAY002	E - COMMERCE	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of this course is to make the students understand

- The scope of E-Commerce in the realm of modern business.
- The technologies used to develop and deliver E-Commerce applications.
- The marketing methods used in E-Commerce
- The legal and regulatory framework in which e-commerce must operate.

Outcomes of the Course:

At the end of the course the student will be able to

- Understand E-Commerce concepts and terminology
- Process management decisions that are involved in launching, operating and managing business activity on the World Wide Web.
- Become familiar with important business, legal issues.

1. INTRODUCTION **6**

Networks and Commercial Transactions - Internet and Other Novelties - Electronic Transactions Today - Commercial Transactions - Establishing Trust - Internet Environment - Internet Advantage - World Wide Web.

2. SECURITY TECHNOLOGIES **9**

Why Internet Is Unsecure - Internet Security Holes - Cryptography : Objective - Codes and Ciphers - Breaking Encryption Schemes - Data Encryption Standard - Trusted Key Distribution and Verification - Cryptographic Applications - Encryption - Digital Signature - Nonrepudiation and Message Integrity.

3. ELECTRONIC PAYMENT METHODS **9**

Traditional Transactions : Updating - Offline and Online Transactions - Secure Web Servers - Required Facilities - Digital Currencies and Payment Systems - Protocols for the Public Transport - Security Protocols - SET - Credit Card Business Basics.

4. ELECTRONIC COMMERCE PROVIDERS **9**

Online Commerce Options - Functions and Features - Payment Systems : Electronic, Digital and Virtual Internet Payment System - Account Setup and Costs - Virtual Transaction Process - InfoHaus - Security Considerations – CyberCash: Model - Security - Customer Protection - Client Application - Selling through CyberCash.

5. ONLINE COMMERCE ENVIRONMENTS **12**

Servers and Commercial Environments - Payment Methods - Server Market Orientation - Netscape Commerce Server - Microsoft Internet Servers - Digital Currencies - DigiCash - Using Ecash - Ecash Client Software and Implementation - Smart Cards - The Chip - Electronic Data Interchange - Internet Strategies, Techniques and Tools.

Total: 45

TEXT BOOK :

1. Pete Loshin, “Electronic Commerce”, 4th Edition, Firewall media, An imprint of laxmi publications Pvt. Ltd., New Delhi, 2004.

REFERENCES :

1. Jeffrey F.Rayport and Bernard J. Jaworski, “Introduction to E-Commerce”, 2nd Edition, Tata Mc-Graw Hill Pvt., Ltd., 2003.

2. Greenstein, “Electronic Commerce”, Tata Mc-Graw Hill Pvt., Ltd., 2000.

CAY003	ADVANCED DATABASES	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of this course is to make the students to

- Use concepts and DBMS features learned previously
- Be familiar with data modeling and able to apply the techniques to medium-complexity problems
- Be proficient with basic SQL and familiar with advanced usage
- Be exposed to database administration
- Be familiar with OBDC and Web site use of databases

Outcomes of the Course:

On successful completion of this course, students will:

- Describe the basic concepts of Relational Database Design
- Explain Database implementation and tools
- Describe SQL and Database System catalog.
- Describe the process of DB Query processing and evaluation.
- Discuss the concepts of transaction management.
- Explain the Database Security and Authorization.
- Describe the design of Distributed Databases.
- Know how to design with DB and XML.
- Describe the basic concept of Data warehousing and Data mining
- Discuss the emerging Database Models Technologies and Applications
- Be able to design high-quality relational databases and database applications.
- Have developed skills in advanced visual & conceptual modeling and database design.
- Be able to translate complex conceptual data models into logical and physical database designs.

1. RELATIONAL DATABASES **9**

Relational Model - Querying - Storage Structures - Query Processing - Normalization.

2. OBJECT ORIENTED DATABASES **9**

Introduction to Object Oriented Data Bases - Approaches - Modeling and Design - Persistence - Transaction - Concurrency - Recovery - Database Administration.

3. EMERGING SYSTEMS **9**

Enhanced Data Models - Client/Server Model - Data Warehousing and Data Mining - Web Databases – Mobile Databases.

4. CURRENT ISSUES **9**

Rules - Knowledge Bases - Active and Deductive Databases - Distributed Databases and Parallel databases.

5. DATABASE DESIGN ISSUES **9**

Security - Integrity - Consistency - Database Tuning - Optimization and Research Issues.

Total: 45

TEXT BOOK :

1. R. Elmasri and S.B. Navathe, Fundamentals of Database Systems, Addison Wesley, 2000.

REFERENCES :

1. Gary W. Hanson and James V. Hanson, Database Management and Design, Prentice Hall of India Pvt Ltd, 1999.

2. Alex Benson, Stephen Smith and Kurt Thearling, Building Data Mining Applications for CRM, Tata McGraw-Hill, 2000.

CAY004	DIGITAL IMAGE PROCESSING	L	T	P	C
		3	0	0	3

Objective of the Course:

The objectives of this course are to

- Cover the basic theory and algorithms that are widely used in digital image processing
- Expose students to current technologies and issues that are specific to image processing systems
- Develop hands-on experience in using computers to process images
- Develop critical thinking about shortcomings of the state of the art in image processing

Outcomes of the Course:

The student will attain the following outcomes:

- Competent in describing discrete-time image signals and systems
- Understand the frequency domain description of discrete-time image processing
- Understand the design & specification of multi-dimensional digital filters for image processing applications
- Understand different approaches to the design and implementation of digital filters
- Gain proficiency in using simulation and design software tools, such as those found in Matlab
- Aware of contemporary applications of image processing.

1. DIGITAL IMAGE FUNDAMENTALS **9**

Image formation, Image transforms – fourier transforms, Walsh, Hadamard, Discrete cosine, Hotelling transforms.

2. IMAGE ENHANCEMENT & RESTORATION **9**

Histogram modification techniques - Image smoothening - Image Sharpening - Image Restoration - Degradation Model – Noise models - Spatial filtering – Frequency domain filtering.

3. IMAGE COMPRESSION & SEGMENTATION 9

Compression Models - Elements of information theory - Error free Compression -Image segmentation –Detection of discontinuities - Edge linking and boundary detection - Thresholding – Region based segmentation - Morphology.

4. REPRESENTATION AND DESCRIPTION 9

Representation schemes- Boundary descriptors- Regional descriptors - Relational Descriptors

5. OBJECT RECOGNITION AND INTERPRETATION 9

Patterns and pattern classes - Decision-Theoretic methods - Structural methods.

Total: 45

TEXT BOOK :

1. Gonzalez.R.C & Woods. R.E., Digital Image Processing, II Ed., Pearson Education, 2002.

REFERENCES :

1. Anil Jain.K, Fundamentals of Digital image Processing, Prentice Hall of India, 1989.

2. Sid Ahmed, Image Processing, McGraw Hill, New York, 1995.

CAY005	TCP/IP PROTOCOL SUITE	L	T	P	C
		3	0	0	3

Objective of the Course:

To provide the skills necessary for the students to design, build and test a small TCP/IP Network, comprising of three subnetworks and two routers. The students will learn how to trace and rectify faults on the network.

Outcomes of the Course:

At the end of the course the student will be able to:

- Describe the four layers of the TCP/IP Protocol suite and the functions of each layer
- Identify the needs and the purpose of each of the protocols at each layer
- Use and configure each of the common applications used with TCP/IP
- Set up and configure a small network
- Configure a router using static routing and RIP
- Split a network into three subnets
- Use the common network commands to troubleshoot a TCP/IP Network
- Examine a TCP/IP trace at all levels, and diagnose network problems

1. INTRODUCTION **10**

Standards – Internet – History- OSI model – Protocol suite – Addressing – Transmission media – Local Area and Wide Area Networks – Switching – Connecting devices – IP addressing.

2. INTERNET PROTOCOL **10**

Subnetting – Supernetting – IP packets – Delivery – Routing – Routing model – Routing table – Datagram – Fragmentation – Checksum – IP Design – ARP – RARP – Internet control message protocol – Internet group management protocol.

3. TRANSMISSION CONTROL PROTOCOL **8**

User Datagram protocol – UDP operation – Use – UDP design – TCP services – Flow control – Error control – TCP operation and design – connection – Transition diagram – Congestion control

4. APPLICATION LAYER AND CLIENT SERVER MODEL 8

Concurrency – BOOTP – DHCP – Domain name system – Name space – Distribution – Resolution – Messages – Telnet – Rlogin – Network Virtual Terminal – Character Set – Controlling the server – Remote login

5. APPLICATION PROTOCOLS 9

File Transfer Protocol – Connections – Communication – Simple Mail Transfer Protocol – Simple Network Management Protocol – Hyper Text Transfer Protocol – Transaction – Request and Response messages.

Total: 45

TEXT BOOK :

1. Behrouz A. Forouzan, "TCP/IP Protocol Suite", Tata McGraw Hill Edition 2000.

REFERENCE :

1. Douglas E. Comer, David L. Stevens, "Internetworking with TCP/IP – Volume I, II and III", Prentice- Hall of India Pvt. Ltd., 2nd Edition 1994

CAY006	VISUAL PROGRAMMING	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is

- To introduce the concepts of visual programming.
- To introduce GUI programming using Microsoft foundation classes.
- To enable the students to develop programs and simple application using Visual C++.

Outcomes of the Course:

The student will

- Demonstrate fundamental skills in utilizing the tools of a visual programming studio environment in terms of the set of available command menus and toolbars.
- Combine event-driven programming with procedural programming.
- Design practical visual forms for business and scientific problem solving applications
- Solve mathematical, scientific, and business problems using visual component based programming.
- Semonstrate skills in "database connectivity" by embedding SQL code in their programs to manipulate external databases.
- Apply visual programming to software creation by designing projects with menus and submenus.
- Use visual programming to create simple computer games.

1. INTRODUCTION **8**

GUI Concepts – Overview of Windows programming – Creating the window - Displaying the window - message Loop – windows procedure-WM_PAINT message - WM_DESTROY message – An Introduction to GDI – Scroll Bars – Keyboard – Mouse – Menus.

2. VISUAL BASIC PROGRAMMING **9**

IDE – First Visual Basic Program - Introduction to Forms –Intrinsic Controls – working with Files - Accessing databases with data control - Classes and Objects – ADO Object Model.

3. VISUAL C++ PROGRAMMING **9**

Windows Programming Model - Visual C++ components – Microsoft foundation classes Library Application Framework – Getting Started with AppWizard – Basic Event handling, Mapping modes, and a Scrolling View - Graphics Device Interface, Colors and fonts – Modal Dialog and Windows Common Dialogs – Modeless Dialog and windows Common dialogs – Using ActiveX controls – Windows Message Processing and Multithreading.

4. ADVANCED CONCEPTS **9**

Menus – Keyboard Accelerators – Rich Edit Control – Tool bars – Status bars – A reusable Frame Window Base Class - Reading and writing documents - SDI and MDI environments – splitter windows and multiple views.

5. APPLICATIONS OF WINDOWS PROGRAMMING **10**

Dynamic link library – Component Object Model - Object linking and embedding – Data Base Management With Microsoft ODBC.

Total: 45

TEXT BOOKS :

1. Charles Petzold, “Windows Programming”, Microsoft press, 1996. Chapters: 2,5,6,9,10.

2. Francesco Balena, “Programming Microsoft Visual Basic6.0”, Microsoft press, Indian Reprint, 2001. Chapters: 1,2,3,5,6,13.

3. David Kruglirski.J, “Programming Microsoft Visual C++”, Fifth Edition, Microsoft press, 1998.

   Chapters: 1,2,3,4,5,6,7,8,12,13,14,15,17,18,20,22,24,31.

REFERENCES :

1. G.Cornell, “Visual Basic 6”, Tata McGraw Hill, 1998.

2. Deitel & Deitel, T.R.Nieto, “Visual Basic 6, How to program”, Prentice Hall of India, 1999.

CAY007	MOBILE COMPUTING	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is to

- Understand the basics of wireless and mobile communications.
- Realize various wireless telecommunication and satellite system
- Know various wireless LAN technologies like ieee 802.11, Bluetooth, etc.,
- Study the issues in mobile routing and transport and also to learn the existing solutions.
- Study Wireless Application layer Protocol (WAP)

Outcomes of the Course:

This course will provide the basic understanding of wireless and mobile communication. It will also provide coverage on wireless telecommunication technologies (GSM, DECT, TETRA, UMTS etc) and satellite systems. This course will deal with the basic architecture and the protocols of wireless LAN like IEEE 802.11, Hiperlan, Bluetooth, etc. This subject also discusses about mobile networks and transport layers. Finally the students will learn about wireless application protocol and its need and usage.

1. INTRODUCTION **9**

Medium Access Control : Motivation for Specialized MAC- SDMA- FDMA- TDMA- CDMA- Comparison of Access mechanisms – Tele communications : GSM- DECT- TETRA – UMTS- IMT-200 – Satellite Systems: Basics- Routing- Localization- Handover- Broadcast Systems: Overview – Cyclic Repetition of Data- Digital Audio Broadcasting – Digital Video Broadcasting

2. WIRELESS NETWORKS **9**

Wireless LAN: Infrared Vs Radio Transmission – Infrastructure Networks- Ad hoc Networks- IEEE 802.11 – HIPERLAN – Bluetooth- Wireless ATM: Working Group- Services- Reference Model – Functions – Radio Access Layer – Handover- Location Management- Addressing Mobile Quality of Service- Access Point Control Protocol

3. MOBILE NETWORK LAYER **9**

Mobile IP : Goals – Assumptions and Requirement – Entities – IP packet Delivery- Agent Advertisement and Discovery – Registration – Tunneling and

Encapsulation – Optimization – Reverse Tunneling – IPv6 – DHCP- Ad hoc Networks

4. MOBILE TRANSPORT LAYER **9**

Traditional TCP- Indirect TCP- Snooping TCP- Mobile TCP- Fast retransmit/ Fast Recovery- Transmission/ Timeout Freezing – Selective Retransmission- Transaction Oriented TCP

5. WAP **9**

Architecture – Datagram Protocol- Transport Layer Security- Transaction Protocol- Session Protocol- Application Environment-Wireless Telephony Application.

Total: 45

TEXT BOOK :

1. J.Schiller, Mobile Communication, Addison Wesley, 2000.

REFERENCES :

1. William C.Y.Lee, Mobile Communication Design Fundamentals, John Wiley, 1993.
2. William Stallings, Wireless Communication and Networks, Pearson Education, 2003.
3. Singhal, WAP-Wireless Application Protocol, Pearson Education, 2003.

CAY008 SOFTWARE QUALITY MANAGEMENT L T P C
3 0 0 3

Objective of the Course:

The objective of the course is to make students aware about the importance of the software testing during software development. The course covered to be in line with the development tools and languages taught in this level. The course will prepare the student for software testing and debugging. It will further lay down the foundation for advanced courses in Software quality assurances.

Outcomes of the Course:

On completion of the course, the student

- Knows and applies quality models to identify and specify the quality attributes a software system must satisfy
- Knows and determines the most appropriate verification and validation techniques to be applied in a software development project with the aim of assuring the quality level required
- Able to identify and determine the practices needed to manage a software system configuration
- Understands the mission of a quality system and knows the applicable standards and norms
- Understands the interrelation between product quality and process quality
- Knows and applies product and process quality control techniques

1. INTRODUCTION 9

Software Process assessment overview - Assessment phases - Assessment principles - Assessment conduct -Implementation consideration - Quality management - Quality assurance plan - Considerations – Verification and Validation.

2. CONFIGURATION MANAGEMENT 9

Need for configuration Management - Software product nomenclature - configuration management functions - Baselines - Responsibilities - Need for automated tools - plan – SCM support functions - The requirement phase Design control - The implementation phase - Test phase - SCM Tools - Configuration accounting and audit.

3. SOFTWARE STANDARDS AND INSPECTION 9

Definitions - Reason for software standards - Benefits - Establishing standards - Guidelines - Types of reviews - Inspection of objectives - Basic inspection principles - The conduct of inspection - Inspection training.

4. TESTING AND MANAGING SOFTWARE QUALITY 9

Testing: principles - Types - Planning - Development - Execution and reporting – Tools and methods - Real Time testing - quality management paradigm - Quality motivation – Measurement criteria - Establishing a software quality program - Estimating software quality.

5. DEFECT PREVENTION 9

Principles of software defect prevention - Process changes for defect prevention - Defect prevention considerations - Managements role - Framework for software process change - Managing resistance to software process change - Case studies.

Total: 45

TEXT BOOK :

1. Watts S. Humphrey, Managing the software process, Addison Wesley, 1999.

REFERENCES :

1. Tsum S.Chow, Software Quality Assurance a Practical Approach, IEEE Computer Society press, 1985.

2. Richard E. Fairley, Software Engineering - A Practitioner's approach, McGraw Hill, 1982.

CAY009 DATA MINING AND DATA WAREHOUSING

L	T	P	C
3	0	0	3

Objective of the Course:

This Course has been designed to provide students with enough knowledge to extract data from large databases. The different algorithms provide knowledge to the student regarding the process of Knowledge discovery & data mining which is one of the emerging fields of Information Technology.

Outcomes of the Course:

On completion of the course, the students will be able to

- Make more effective use of data stored in databases
- Create a clean, consistent repository of data within a data warehouse
- Utilise various levels and types of summarisation of data to support management decision making.
- Discover patterns and knowledge that is embedded in the data using several different techniques, such as neural nets, decision trees and statistical techniques.

1. INTRODUCTION **9**

Relation To Statistics, Databases- Data Mining Functionalities-Steps In Data Mining Process-Architecture Of A Typical Data Mining Systems- Classification Of Data Mining Systems - Overview Of Data Mining Techniques.

2. DATA PREPROCESSING AND ASSOCIATION RULES **9**

Data Preprocessing-Data Cleaning, Integration, Transformation, Reduction, Discretization Concept Hierarchies-Concept Description: Data Generalization And Summarization Based Characterization- Mining Association Rules In Large Databases.

3. PREDICTIVE MODELING **9**

Classification And Prediction: Issues Regarding Classification And Prediction-Classification By Decision Tree Induction-Bayesian Classification-Other Classification Methods-Prediction-Clusters Analysis: Types Of Data In Cluster Analysis- Categorization Of Major Clustering Methods: Partitioning Methods –Hierarchical Methods

4. DATA WAREHOUSING **9**

Data Warehousing Components -Multi Dimensional Data Model- Data Warehouse Architecture-Data Warehouse Implementation- -Mapping the Data Warehouse to Multiprocessor Architecture- OLAP.-Need- Categorization of OLAP Tools.

5. APPLICATIONS **9**

Applications of Data Mining-Social Impacts Of Data Mining-Tools-An Introduction To DB Miner-Case Studies-Mining WWW-Mining Text Database-Mining Spatial Databases.

Total: 45

TEXT BOOK :

1. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2002.

REFERENCES :

1. Alex Berson,Stephen J. Smith, “Data Warehousing, Data Mining,& OLAP”, Tata McGraw- Hill, 2004.
2. Usama M.Fayyad, Gregory Piatetsky - Shapiro, Padhrai Smyth And Ramasamy Uthurusamy, "Advances In Knowledge Discovery And Data Mining", The M.I.T Press, 1996.
3. Ralph Kimball, "The Data Warehouse Life Cycle Toolkit", John Wiley & Sons Inc., 1998.
4. Sean Kelly, "Data Warehousing In Action", John Wiley & Sons Inc., 1997.

CAY010 COMPONENT BASED TECHNOLOGY

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is

- To introduce the fundamental properties of components, technology and architecture and middleware.
- To give exposure to java based component technologies
- To impart knowledge on component technologies such as CORBA, ORB and application server.
- To introduce COM, DCOM and .NET technologies.
- To impart knowledge in component frameworks and development.

Outcomes of the Course:

On completion of the course, students will

- Be familiar with the principles of integrating and exchanging data
- Master the use of XML/DOM to integrate and exchange data, and XSL/ XSLT to transform data
- Be familiar with principles for building software systems from components, including service-oriented architecture and appropriate design patterns
- Be familiar with evidence-based security and code access security
- Be familiar with the functions of a software version control system
- Be familiar with building server applications that consume web services, exchange, and transform web-based data

1. INTRODUCTION **8**

Definition - Industrialization of software development - CBD drivers and benefits - Technology evolution - Components and network computing.

2. FUNDAMENTALS **10**

Basic concepts of CBD - Scenarios for CBD - Evolution or revolution - Build,find and use components and objects.

3. MODELS **10**

Basic concepts of object models - Components and interfaces - Working with

interfaces - Component and interface modeling - Specification models - domain modeling - Describing classes - Patterns and frameworks.

4. USING CBD **9**

Categorizing & deploying components - CORBA, DCOM.

5. FRAMEWORKS **8**

Class libraries - Encapsulated components - Software frameworks - Pre - built applications.

Total: 45

TEXT BOOK :

1. Kuth Short, Component Based Development and Object Modeling, Sterling software,1997.

REFERENCE :

1. Clemens Szyperski, Component software - Beyond object - Oriented programming, Addison - Wesley, 2000.

CAY011 SUPPLY CHAIN MANAGEMENT L T P C
3 0 0 3

Objective of the Course:

The objective of this course is to understand the finer aspects of the supply chain management that can turnaround an organization and can take it to the position of a leader. The course also aims to learn from the various successes and failures in making an efficient and effective supply chain and deliver value to the customer.

Outcomes of the Course:

At the completion of this course, the student would be able to examine the design and performance of supply networks and processes in different business contexts. Students develop capabilities in logistics, digital coordination for supply chain integration, inventory management; risk pooling, procurement, product and process design, and international supply chain management.

1. BASIC CONCEPTS 9

Introduction to supply chain management (SCM) – concept of SCM – Components of SCM, an overview – features of SCM – strategic issues in SCM – Systems View - SCM current scenario – value chain management and customer relations management.

2. INTERFACES WITH OTHER DISCIPLINES 10

Marketing and Supply Chain Interface – Customer focus in SCM – Demand planning Purchase planning – Make or Buy decision – Indigenous and global sourcing – Development and management of suppliers – legal aspects of buying – cost management – negotiating for purchasing/subcontracting – purchase insurance – evaluation of purchase performance (performance indices).Inventory management.- Finance and Supply Chain Interface. Financial impact of inventory.

3. MANUFACTURING AND WAREHOUSING 9

Manufacturing scheduling – Manufacturing flow system – work flow automation – Flexibility in manufacturing to achieve dynamic optimization. Material handling system design and decision. Warehousing and store keeping – strategies of warehousing and storekeeping – space management.

4. LOGISTICS MANAGEMENT **8**

Logistics management – Role of logistics in SCM – Integrated Logistics management – transportation design and decision – multi modalism – third party logistics services and providers – facilities management (port/ airport.ICD's) channels of distribution – logistics and customer service.

5. INFORMATION TECHNOLOGY AND SCM **9**

Information technology and SCM – EDI, ERP, Internet and Intranet, E-Commerce, Bar coding, Telecommunication Network, Advanced planning system, Decision support models for Supply Chain Management, Artificial Intelligence for SCM- Best practice in supply chain management – organizational issues to implement SCM.

Total: 45

TEXT BOOK :

1. B.S.Sahay, Supply chain management for global competitiveness, Macmillan India Limited, 2000.

REFERENCES :

1. Donald J.Bowersox & David J.Closs, Logistical Management, Tata McGraw-Hill Editions, New Delhi, 2000.
2. David Simchi-Levi, Designing and managing the supply chain, Tata McGraw-Hill Editions, New Delhi, 2000

CAY012	BUSINESS PROCESSES	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is

- To develop a business process strategy to meet stakeholder needs.
- To analyse, improve, design and develop processes to meet stakeholder needs.
- To align technology, organisation, and facilities with the business process strategy and design
- To apply their knowledge to manage process projects effectively.
- To identify, clarify and manage business benefits arising from process change

Outcomes of the Course:

At the completion of this course, the student will be able to form the organizational structure, improve leadership quality, take steps to improve reengineering of Indian scenario, to form organizational frame work and implementation of e-business application areas.

1. ORGANIZATIONAL STRUCTURE **9**

Types of Business Organizations-Organizational Structures-Definition-Complexity-Formulization-Size-Technology-Culture-Forms and Outcomes-Explanations of Structures-IT Industry and Organizational Structures-Case Studies

2. ORGANIZATIONAL OUTCOMES **9**

Organizational Power and Power Outcomes-Leadership and Decision Making-Communication and Organizational Change-Organizational Environments and Effects-Inter and Intra organizational Relationships-Organizational Effectiveness-Case Studies

3. BUSINESS PROCESS RE-ENGINEERING **9**

Introduction to Business Process Re-engineering (BPR)-Meaning-Types-Process-Impetrative for Survival-Strategic Approach-Implementing Business Process Re-engineering-Methodology and Steps-Indian Scenario of Implementing BPR-Case Studies

4. BPR AND IT INDUSTRY **9**

BPR and Information Technology Process-People View and Perspectives-Empowering People through IT-Managing Change in the Global Environment-BPR Rediscovering Indian Paradigm-Need of Reengineering-Case Studies

5. E-BUSINESS PROCESS **9**

E-Business-Introduction-E-business vs. E-commerce-Execution of E-business-Trends-Design for Execution-Construction-Types-Organizational Frame Work and Implementation-E-business Application Areas(CRM,ERP,SCM and Selling)-E-business and India-Case Studies

Total: 45

TEXTBOOKS :

1. Richard H.Hall, Organizations-Structures, Processes and Outcomes", Pearson Education, 2004
2. M.S.Jayaraman et. Al, "Business Process Reengineering", Tata Mc Graw Hill Publications, 2001
3. Ravi Kalakota and Marcia Robinson, "E-Business; Roadmap for Success; Pearson Education, 2000

REFERENCES :

1. Gareth Jones, "Organizational Theory, Design and Change", Pearson Education, 4th Edition, 2004
2. Dave Chaffey, "E-business and E-Commerce" Pearson Education, 2nd Edition, 2003

CAY013 SOFTWARE PROJECT MANAGEMENT

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is

- To understand the concepts of Software project management.
- To know the techniques in developing Quality Software Products
- To Manage the Software Product Development

Outcomes of the Course:

On completion of the course, the students will be able to

- Gain Knowledge to develop Quality Software Products
- Plan, organize and manage the various resources effectively to achieve specific
- Attain goals in the Software Industry.
- Maintain International standards in Software development.
- Handle Software projects effectively.

1. INTRODUCTION **9**

Introduction to Competencies - Product Development Techniques - Management Skills - Product Development Life Cycle - Software Development Process and models - The SEI CMM-International Organization for Standardization.

2. DOMAIN PROCESSES **9**

Managing Domain Processes - Project Selection Models - Project Portfolio Management - Financial Processes - Selecting a Project Team - Goal and Scope of the Software Project - Project Planning - Creating the Work Breakdown Structure - Approaches to Building a WBS - Project Milestones - Work Packages - Building a WBS for Software.

3. SOFTWARE DEVELOPMENT **9**

Tasks and Activities - Software Size and Reuse Estimating - The SEI CMM - Problems and Risks - Cost Estimation - Effort Measures - COCOMO: A Regression Model - COCOMO II - SLIM: A Mathematical Model - Organizational Planning - Project Roles and Skills Needed.

4. SCHEDULING ACTIVITIES **9**

Project Management Resource Activities - Organizational Form and Structure - Software Development Dependencies - Brainstorming - Scheduling Fundamentals - PERT and CPM - Leveling Resource Assignments - Map the Schedule to a Real Calendar - Critical Chain Scheduling.

5. QUALITY ASSURANCE **9**

Quality: Requirements – The SEI CMM - Guidelines - Challenges - Quality Function Deployment - Building the Software Quality Assurance - Plan - Software Configuration Management: Principles - Requirements - Planning and Organizing - Tools - Benefits - Legal Issues in Software - Case Study.

Total: 45

TEXT BOOK :

1. Robert T. Futrell, Donald F. Shafer, Linda I. Safer, “Quality Software Project Management”, Pearson Education, Asia, 2002.

REFERENCES :

1. Pankaj Jalote, “Software Project Management in Practice”, Addison Wesley, 2002.
2. Hughes, “Software Project Management, 3/E”, Tata McGraw-Hill, 2004.

CAY014 ENTERPRISE RESOURCE PLANNING

L	T	P	C
3	0	0	3

Objective of the Course:

The aim of the course is

- To describe the concept of ERP and the ERP model; define key terms; explain the transition from MRP to ERP; identify the levels of ERP maturity.
- To explain how ERP is used to integrate business processes; define and analyze a process; create a process map and improve and/or simplify the process; apply the result to an ERP implementation.
- To describe the elements of a value chain, and explain how core processes relate; identify how the organizational infrastructure supports core business processes; explain the effect of a new product launch on the three core business processes.
- To identify the international issues that impact a worldwide implementation of ERP; identify the key technological considerations and infrastructure concerns in ERP implementation; describe the strategic use of technology for ERP.
- To explain how the key elements of organizational change management apply to an ERP implementation; define change readiness; describe a learning requirements plan; explain the use of assessment tools to identify the readiness of an organization to change; identify the methods of implementing and sustaining the change.
- To describe project organizational considerations; define the project management tools and resources needed to implement an ERP system; describe the roles and responsibilities of the key organization players; describe the tactics, tools, and methodologies available to implement ERP; evaluate the success of the implementation.
- To describe how the knowledge management capability of an ERP system can be used to sustain competitive advantage; describe how to use ERP to communicate effectively with customers and suppliers.

Outcomes of the Course:

A student completing this course will:

- Understand and gain insight into process views of organizations and tools and techniques used to model both as-is and to-be models.

- Apply the process modeling techniques in one or more modeling environments.
- Know and be able to apply key technical terminology in enterprise information systems as they apply in different ERP products and development methods
- Understand key differences between the major ERP applications and issues specific to these applications their configuration and management.
- Analyze a current architecture and perform an effective gap analysis before an ERP implementation
- Be able to map enterprise architectural resources to a contemporary Enterprise Architecture mapping tool.
- Understand and be able to articulate the life cycle stages of any ERP implementation.
- Effectively describe problems typical of ERP implementation projects and translate this information and use this information to anticipate and articulate the challenges associated with post-implementation management of ERP systems.
- Synthesize prior theoretical and experiential knowledge in IT development and project management with the current literature on Enterprise System development.
- Be able to evaluate the progress of an ongoing ERP implementation project.

1. INTRODUCTION TO ERP **9**

Integrated Management Information Seamless Integration – Supply Chain Management – Integrated Data Model – Benefits of ERP – Business Engineering and ERP – Definition of Business Engineering – Principle of Business Engineering – Business Engineering with Information Technology.

2. BUSINESS MODELLING FOR ERP **9**

Building the Business Model – ERP Implementation – An Overview – Role of Consultant, Vendors and Users, Customisation – Precautions – ERP Post Implementation Options-ERP Implementation Technology –Guidelines for ERP Implementaion.

3. ERP AND THE COMPETITIVE ADVANTAGE **9**

ERP domain MPGPRO – IFS/Avalon – Industrial and Financial Systems – Baan IV SAP-Market Dynamics and Dynamic Strategy.

4. COMMERCIAL ERP PACKAGE **9**

Description – Multi-Client Server Solution – Open Technology – User Interface-Application Integration.

5. ARCHITECTURE **9**

Basic Architectural Concepts – The System Control Interfaces – Services – Presentation Interface – Database Interface.

Total: 45

TEXT BOOK :

1. Vinod Kumar Garg and N.K.Venkita Krishnan, “Enterprise Resource Planning – Concepts and Practice”, PHI, 1998.

REFERENCE :

1. Jose Antonio Fernandz, The SAP R/3 Handbook, TMH, 1998.

CAY015	SOFTWARE AGENTS	L	T	P	C
		3	0	0	3

Objective of the Course:

Agent-based systems are software products that not only do things as specified but also have knowledge to do their job and can do it in a cooperative, coordinative and competitive way.

Aim of the course is

- To understand the myths and realities of the agent-based systems?
- To develop an agent-based system for a particular task?
- To evolve from object-oriented development to agent-based systems?
- To incorporate and share knowledge among software agents?

Outcomes of the Course:

After taking this course, the students

- Will have an understanding of the agent system terminology and development process of agent-based systems.
- Will have learned techniques to design agent-based system.
- Will know how to modify architecture of the current software systems and re-structure them to be agent-based.

1. AGENT AND USER EXPERIENCE **9**

Interacting with Agents - Agent From Direct Manipulation to Delegation - Interface Agent Metaphor with Character - Designing Agents - Direct Manipulation versus Agent Path to Predictable

2. AGENTS FOR LEARNING IN INTELLIGENT ASSISTANCE **9**

Agents for Information Sharing and Coordination - Agents that Reduce Work Information Overhead - Agents without Programming Language - Life like Computer character - S/W Agents for cooperative Learning - Architecture of Intelligent Agents.

3. AGENT COMMUNICATION AND COLLABORATION **9**

Overview of Agent Oriented Programming - Agent Communication Language - Agent Based Framework of Interoperability

4. AGENT ARCHITECTURE **9**

Agents for Information Gathering - Open Agent Architecture - Communicative Action for Artificial Agent

5. MOBILE AGENTS **9**

Mobile Agent Paradigm - Mobile Agent Concepts -Mobile Agent Technology - Case Study: Tele Script, Agent Tel

Total: 45

TEXT BOOKS :

1. Jeffrey M.Bradshaw," Software Agents ", MIT Press, 2000. (Unit 1,2,3 & 4)
2. William R. Cockayne, Michael Zyda, “Mobile Agents", Prentice Hall, 1998 (5th Unit)

REFERENCES :

1. Russel & Norvig, " Artificial Intelligence: A Modern Approach ", Prentice Hall, 2nd Edition, 2002
2. Joseph P.Bigus & Jennifer Bigus, “Constructing Intelligent agents with Java: A Programmer's Guide to Smarter Applications ", Wiley, 1997.

CAY016	UNIX INTERNALS	L	T	P	C
		3	0	0	3

Objective of the Course:

This course provides an in depth knowledge of the UNIX operating system's internal features and their operation. The course describes the data structures, their relationships and the major algorithms used to manage System, processes, system calls, interrupts and exceptions, virtual memory and file systems.

Outcomes of the Course:

Students who have successfully completed this course will have full understanding of the following concepts:

- Learn the basic Unix operating systems and its basic commands, the buffer representation, kernels and system calls,the system structure, implementation of system calls, the UNIX segmentation, paging and scheduling and the Drivers and IPC.
- Ability to understand the basic functioning of UNIX operating systems and shell programming, to analyze the buffers and kernel representation, to understand the UNIX system structure, system calls and to understand UNIX segmentation, scheduling, paging.

1. INTRODUCTION TO UNIX **9**

Unix operating system - History - System structure –Users Perspective- OS Services- Hardware-Architecture- System Concepts- Kernel data structures – System Administration – Buffer Cache- Heaters – Structure of the Buffer Pool- Scenarios-Reading and writing Disk Blocks.

2. FILE SYSTEMS **9**

INODES - Structure of a regular file- Directories – Conversion of a path name to an INODE - Super Block- INODE assignment – Disk Blocks- System calls for the file system

3. PROCESSES **9**

Process States and Transitions – Layout of System Memory – Context of a Process – Manipulation of the process address space – Sleep – Process Control – Creation – Signals – Awaiting process termination – The Shell – System Boot and Init Process – Process Scheduling and Time – System calls for time – Clock.

4. MEMORY MANAGEMENT **9**

Swapping – Segmentation - Demand Paging – Driver Interfaces – Disk Drivers – Terminal Drivers - Streams.

5. INTERPROCESS COMMUNICATION **9**

Process Tracing – System V IPC – Network Communications - Sockets – Problem of Multiprocessor Systems – Solution with Master and Slave Processors – Semaphores – Distributed Unix Systems – Satellite Processors – Newcastle connection – Transparent distributed file systems – System Calls.

Total: 45

TEXT BOOK :

1. Bach M.J., The Design of the Unix Operating System, Prentice Hall India, 1986.

REFERENCES :

1. Goodheart B., Cox.J., The Magic Garden Explained, Prentice Hall India, 1994.

2. Leffler S.J., Mckusick M.K., Karels M.J and Quarterman J.S., The Design and Implementation of the 4.3 BSD Unix Operating System. Addison Wesley, 1998.

CAY017	GRID COMPUTING	L	T	P	C
		3	0	0	3

Objective of the Course:

Aim of the course is

- To provide insight into the architectural implications of Grid Computing
- To provide students with awareness of current issues in
- Skills in utilizing current grid tools and technologies.
- Identifying the weakness of existing tools and technologies and proposing potential areas for improvement.
- Justify the applicability, non-applicability of Grid technologies for a specific application

Outcomes of the Course:

On completion of the course,

- The students will attain knowledge in Grid Computing.
- Students will understand the applications of Grid Computing

1. INTRODUCTION **9**

Grid Computing values and risks – History of Grid computing – Grid computing model and protocols – overview of types of Grids.

2. TYPES OF GRIDS **9**

Desktop Grids : Background – Definition – Challenges – Technology – Suitability – Grid server and practical uses; Clusters and Cluster Grids; HPC Grids; Scientific in sight – application and Architecture – HPC application development environment and HPC Grids; Data Grids; Alternatives to Data Grid – Data Grid architecture.

3. ARCHITECTURE AND MANAGEMENT **9**

The open Grid services Architecture – Analogy – Evolution – Overview – Building on the OGSA platform – implementing OGSA based Grids – Creating and Managing services – Services and the Grid – Service Discovery – Tools and Toolkits – Universal Description Discovery and Integration (UDDI)

4. NATIVE PROGRAMMING AND SOFTWARE APPLICATIONS 9

Desktop supercomputing – parallel computing – parallel programming paradigms – problems of current parallel programming paradigms – Desktop supercomputing programming paradigms – parallelizing existing applications – Grid enabling software applications – Needs of the Grid users – methods of Grid deployment – Requirements for Grid enabling software – Grid enabling software applications.

5. APPLICATIONS, SERVICES AND ENVIRONMENTS 9

Application integration – application classification – Grid requirements – Integrating Applications with Middleware platforms – Grid enabling Network services – managing Grid environments – Managing Grids – Management reporting – Monitoring – Data catalogs and replica management – portals – Different application areas of Grid computing.

Total: 45

TEXT BOOK:

1. Ahmar Abbas, “Grid Computing, A Practical Guide to Technology and Applications”, Firewall media , 2004.

REFERENCE :

1. Joshy Joseph, Craig Fellenstein, “Grid Computing”, Pearson Education, 2004. Foster, “Grid Blue print foe new computing”.

CAY018	NETWORK SECURITY	L	T	P	C
		3	0	0	3

Objective of the Course:

The aim of the course is

- To acquire an understanding of network security and its changing character
- To understand how network security is conceptualized and carried out
- To examine the historical evolution of network security
- To analyze both early and contemporary threats to network security
- To articulate informed opinion about issues related to network security
- To identify and investigate threats to network security
- To appreciate the challenges of network security

Outcomes of the Course:

Completion of the course will have following outcomes:

- Development of the student's analytical thinking capabilities through comparison and contrast in the application of theories and concepts to social problems
- Enhancement of the student's literacy skills through assigned readings, discussions, examinations, and other requirements
- Utilization of the historical perspective which helps with student understanding of evolutionary developments over time
- Understanding of the scientific method through research requirements and in-depth case studies
- Improvement in social interaction skills and understanding human behavior through consideration of the impact legal and social systems have on individuals
- Increased student awareness of cultural and multi-cultural issues through study of how social problems and social movements are related to, and affect, minority groups
- Preparation for more advanced study in criminal justice/homeland security

1. INTRODUCTION **9**

Attacks - Services - Mechanisms - Conventional Encryption - Classical And Modern Techniques – Encryption Algorithms - Confidentiality.

2. PUBLIC KEY ENCRYPTION **9**

RSA - Elliptic Curve Cryptography - Number Theory Concepts

3. MESSAGE AUTHENTICATION **9**

Hash Functions - Digest Functions - Digital Signatures - Authentication Protocols.

4. NETWORK SECURITY PRACTICE **9**

Authentication, Applications - Electronic Mail Security - IP Security - Web Security.

5. SYSTEM SECURITY **9**

Intruders – Viruses – Worms – Firewalls Design Principles – Trusted Systems.

Total: 45

TEXT BOOK :

1. Stallings, Cryptography & Network Security - Principles & Practice, Prentice Hall, 3rd Edition 2002.

REFERENCES:

1. Bruce, Schneier, Applied Cryptography, 2nd Edition, Toha Wiley & Sons, 1996.
2. Man Young Rhee, “Internet Security”, Wiley, 2003.
3. Pfleeger & Pfleeger, “Security in Computing”, Pearson Education, 3rd Edition, 2003.

CAY019	EMBEDDED SYSTEMS	L	T	P	C
		3	0	0	3

Objective of the Course:

The purpose of the Embedded System course is to provide the students, knowledge and hands-on experience in the embedded computer system technology. The course is intended to students specializing in software and system design, giving most attention to hardware design, as understanding in this area is vital for most embedded software designers. In short, this course is a chance for the very novices in electronics design to create a simple embedded computer based device and to learn the maximum from the experience.

Outcomes of the Course:

On completion of the course:

- Students get exposure with different families and architectures of Embedded System tools such as Microcontrollers, DSPs, FPGAs etc.
- Students shall get the expertise required to design any embedded system (h/w or s/w or both) based on any of the above tools
- Students become highly proficient in Embedded Software particularly in real-time programming with Industry standard RTOS such as VxWorks and RTLinux.

1. INTRODUCTION TO EMBEDDED SYSTEMS 9

Definition and Classification – Overview of Processors and hardware units in an embedded system – Software embedded into the system – Exemplary Embedded Systems – Embedded Systems on a Chip (SoC) and the use of VLSI designed circuits.

2. DEVICES AND BUSES FOR DEVICES NETWORK 9

I/O Devices - Device I/O Types and Examples – Synchronous - Iso-synchronous and Asynchronous Communications from Serial Devices - Examples of Internal Serial-Communication Devices - UART and HDLC - Parallel Port Devices - Sophisticated interfacing features in Devices/Ports-Timer and Counting Devices - ‘12C’, ‘USB’, ‘CAN’ and advanced I/O Serial high speed buses- ISA, PCI, PCI-X, cPCI and advanced buses.

3. EMBEDDED PROGRAMMING **9**

Programming in assembly language (ALP) vs. High Level Language - C Program Elements, Macros and functions -Use of Pointers - NULL Pointers - Use of Function Calls – Multiple function calls in a Cyclic Order in the Main Function Pointers – Function Queues and Interrupt Service Routines Queues Pointers – Concepts of EMBEDDED PROGRAMMING in C++ - Objected Oriented Programming – Embedded Programming in C++, 'C' Program compilers – Cross compiler – Optimization of memory codes.

4. REAL TIME OPERATING SYSTEMS – PART - 1 **9**

OS Services – Interrupt Routines Handling, Task scheduling models - Handling of task scheduling and latency and deadlines as performance metrics - Inter Process Communication And Synchronisation – Shared data problem – Use of Semaphore(s) – Priority Inversion Problem and Deadlock Situations – Inter Process Communications using Signals – Semaphore Flag or mutex as Resource key – Message Queues – Mailboxes – Pipes – Virtual (Logical) Sockets – RPCs.

5. REAL TIME OPERATING SYSTEMS – PART - 2 **9**

Study of RTOS, VxWorks - Basic Features - Task Management Library at the System - Library Header File - VxWorks System Functions and System Tasks - Inter Process (Task) Communication Functions - Case Study of Coding for Sending Application Layer Byte Streams on a TCP/IP Network Using RTOS Vxworks.

Total : 45

TEXT BOOK :

1. Rajkamal, Embedded Systems Architecture, Programming and Design, TATA McGraw Hill, First reprint 2003

REFERENCE :

1. David E.Simon, An Embedded Software Primer, Pearson Education Asia, First Indian Reprint 2000.

CAY020 SOFTWARE QUALITY ASSURANCE

L	T	P	C
3	0	0	3

Objective of the Course:

This course equips the students with a solid understanding of:

- Practices that support the production of quality software
- Software testing techniques
- Life-cycle models for requirements, defects, test cases, and test results
- Process models for units, integration, system, and acceptance testing
- Quality Models

Outcomes of the Course:

At the completion of this course, the student will be able to:

- Understand and effectively apply software quality assurance (SQA) methods, tools and techniques
- Plan for, implement and manage the integrated software quality assurance function
- Provide the necessary software quality assurance steps, controls and results needed at each step or phase of the system's development life cycle to assure communication and satisfaction with both user/client and information systems personnel
- Evaluate how new technologies impact software quality assurance and the system's development life cycle and understand how to benefit from their application

1. CONCEPTS **9**

Concepts of Quality Control, Quality Assurance, Quality Management - Total Quality Management; Cost of Quality; QC tools - 7 QC Tools and Modern Tools; Other related topics - Business Process Re-engineering –Zero Defect, Six Sigma, Quality Function Deployment, Benchmarking, Statistical process control.

2. SOFTWARE ENGINEERING CONCEPTS **9**

Software Engineering Principles, Software Project Management, Software Process, Project and Product Metrics, Risk Management, Software Quality Assurance; Statistical Quality Assurance - Software Reliability, Muse Model;

Software Configuration Management; Software Testing; CASE (Computer Aided Software Engineering).

3. QUALITY ASSURANCE MODELS **9**

Models for Quality Assurance-ISO-9000 - Series, CMM, SPICE, Malcolm Baldrige Award.

4. SOFTWARE QUALITY ASSURANCE RELATED TOPICS **9**

Software Process - Definition and implementation; internal Auditing and Assessments; Software testing -Concepts, Tools, Reviews, Inspections & Walkthroughts; P-CMM.

5. FUTURE TRENDS **9**

PSP and TSP, CMMI, OO Methodology, Clean-room software engineering, Defect injection and prevention.

Total: 45

TEXT BOOK :

1. Watts Humphery, “Managing Software Process", Addison - Wesley, 1998.

REFERENCES:

1. Philip B Crosby, "Quality is Free: The Art of Making Quality Certain", Mass Market, 1992.
2. Roger Pressman, “Software Engineering", Sixth Edition, McGraw Hill, 2005.

CAY021	ADHOC NETWORKS	L	T	P	C
		3	0	0	3

Objective of the Course:

The course examines wireless cellular, ad hoc and sensor networks, covering topics such as wireless communication fundamentals, medium access control, network and transport protocols, unicast and multicast routing algorithms, mobility and its impact on routing protocols, application performance, quality of service guarantees, and security. Energy efficiency and the role of hardware and software architectures will also be presented for sensor networks.

Outcomes of the Course:

At the completion of this course, it is envisaged that the students will be able to:

- Explain the constraints of the wireless physical layer that affect the design and performance of ad hoc and sensor networks, protocols, and applications;
- Explain the performance of various unicast and multicast routing protocols that have been proposed for ad hoc networks;
- Explain the operation of several media access protocols that have been proposed for ad hoc and sensor networks;
- Describe the platform architectures that are suitable for mobile computing and communications, e.g. personal digital assistants (PDAs), handsets, etc.;
- Explain the energy issues in sensor networks and how they can be addressed using scheduling, media access control, and special hardware;
- Explain various security threats to ad hoc networks and describe proposed solutions.

1. INTRODUCTION **9**

Introduction-Fundamentals of Wireless Communication Technology - The Electromagnetic Spectrum - Radio Propagation Mechanisms - Characteristics of the Wireless Channel - IEEE 802.11a,b Standard – Origin Of Ad hoc: Packet Radio Networks - Technical Challenges - Architecture of PRNETs - Components of Packet Radios – Ad hoc Wireless Networks -What Is an Ad Hoc Network? Heterogeneity in Mobile Devices - Wireless Sensor Networks - Traffic Profiles - Types of Ad hoc Mobile Communications - Types of Mobile Host Movements - Challenges Facing Ad Hoc Mobile Networks-Ad hoc wireless Internet

2. ADHOC ROUTING PROTOCOLS 9

Introduction - Issues in Designing a Routing Protocol for Ad Hoc Wireless Networks - Classifications of Routing Protocols -Table-Driven Routing Protocols - Destination Sequenced Distance Vector (DSDV) - Wireless Routing Protocol (WRP) - Cluster Switch Gateway Routing (CSGR) - Source-Initiated On-Demand Approaches - Ad Hoc On-Demand Distance Vector Routing (AODV) - Dynamic Source Routing (DSR) -Temporally Ordered Routing Algorithm (TORA) - Signal Stability Routing (SSR) -Location-Aided Routing (LAR) - Power-Aware Routing (PAR) - Zone Routing Protocol (ZRP)

3. MULTICASTROUTING IN ADHOC NETWORKS 9

Introduction - Issues in Designing a Multicast Routing Protocol - Operation of Multicast Routing Protocols - An Architecture Reference Model for Multicast Routing Protocols -Classifications of Multicast Routing Protocols - Tree-Based Multicast Routing Protocols- Mesh-Based Multicast Routing Protocols - Summary of Tree-and Mesh-Based Protocols - Energy-Efficient Multicasting - Multicasting with Quality of Service Guarantees - Application-Dependent Multicast Routing - Comparisons of Multicast Routing Protocols

4. TRANSPORT LAYER, SECURITY PROTOCOLS 9

Introduction - Issues in Designing a Transport Layer Protocol for Ad Hoc Wireless Networks - Design Goals of a Transport Layer Protocol for Ad Hoc Wireless Networks -Classification of Transport Layer Solutions - TCP Over Ad Hoc Wireless Networks -Other Transport Layer Protocols for Ad Hoc Wireless Networks - Security in Ad Hoc Wireless Networks - Network Security Requirements - Issues and Challenges in Security Provisioning - Network Security Attacks - Key Management - Secure Routing in Ad Hoc Wireless Networks

5. QoS AND ENERGY MANAGEMENT 9

Introduction - Issues and Challenges in Providing QoS in Ad Hoc Wireless Networks -Classifications of QoS Solutions - MAC Layer Solutions - Network Layer Solutions - QoS Frameworks for Ad Hoc Wireless Networks Energy Management in Ad Hoc Wireless Networks –Introduction - Need for Energy Management in Ad Hoc Wireless Networks - Classification of Energy Management Schemes - Battery Management Schemes - Transmission Power Management Schemes - System Power Management Schemes

TEXT BOOK :

1. C. Siva Ram Murthy and B.S. Manoj “Ad Hoc Wireless Networks: Architectures and Protocols”, Prentice Hall PTR,2004

REFERENCES:

1 C.K. Toh, Ad Hoc Mobile Wireless Networks: Protocols and Systems, Prentice Hall PTR ,2001

2. Charles E. Perkins, Ad Hoc Networking, Addison Wesley, 2000

CAY022	SPECIAL ELECTIVE	L	T	P	C
		3	0	0	3