

BHARAI VIDYAPEETH UNIVERSITY, PUNE

FACULTY OF MANAGEMENT STUDIES

Board of Studies in Computer Applications and Systems Studies

Structure of Master of Computer applications Programme

(Under Choice Based Credit System)

To be effective from 2011-12 at Part I

1. INTRODUCTION

The Master of Computer Applications (MCA) Program is a full time 150 Credits program offered by Bharati Vidyapeeth University (BVU), Pune and conducted at its management institutes in Karad, Kolhapur, Pune, Sangli, and Solapur. All the five institutes have excellent faculty, Laboratories, Library, and other facilities to provide proper learning environment. The five institutes are approved by AICTE to conduct MCA program, while the University itself is accredited by NAAC with an 'A' grade. The expectations and requirements of the Software Industry, immediately and the near future, are visualized while designing the MCA program. This effort is reflected in the Vision and Mission statements of the MCA program. Of course, the statements also embody the spirit of the vision of Dr. Patangraoji Kadam, the Founder of Bharati Vidyapeeth and Chancellor, Bharati Vidyapeeth University which is to usher in “Social Transformation through Dynamic Education.”

2. VISION STATEMENT OF MCA PROGRAM

To create high caliber solution architects and innovators for software development

3. MISSION STATEMENT OF MCA PROGRAM

To teach 'things just not words', 'how to think', and 'how to self-learn'.

4. OBJECTIVES OF THE MCA PROGRAM

The main objectives of MCA Program are to prepare the youth to take up positions as system analysts, system engineers, software engineers, programmers and of course as versatile teachers in any area of computer applications. Accordingly the course curriculum aims at developing 'systems thinking' 'abstract thinking', 'skills to analyze and synthesize', and 'skills to apply knowledge', through 'extensive problem solving sessions', 'hands on practice under various hardware/software environments', 'four minor projects and 'one semester full-time internship project'. In addition, 'social interaction skills', 'communication skills', 'life skills', 'entrepreneurial skills', and 'research skills' which are necessary for career growth and for leading quality life are also imparted.

5. LEARNING OUTCOMES FROM THE MCA PROGRAM

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

- (a) Analyze problems and design effective and efficient software solutions
- (b) Develop software under latest Application Development Environments.
- (c) Learn new technologies with ease and be productive at all times
- (d) Read, write, and contribute to technical literature
- (e) Work in teams
- (f) Be a good citizen in all respects.

6. ELIGIBILITY FOR ADMISSION TO THIS COURSE

Admission to the course is open to any graduate (10+2+3) of any recognized university satisfying the following conditions.

The candidate should have secured at least 50% (45% for SC/ST) in aggregate at graduate level university examination.

The knowledge of mathematics / Statistics at XII level or higher level is essential except for BCA, B.Sc. (Computer), BCS graduates.

Subject to the above conditions, the final admission is based solely on the merit at the all India entrance test (B-CAT) conducted by Bharati Vidyapeeth University, Pune.

7. DURATION OF THE COURSE

The duration of this course is three years divided in to six semesters or a minimum of 150 credits whichever is later. The medium of instruction and examination will be only English.

8. STANDARD OF PASSING:

A) The 7-point scale would be used to convert marks out of 100 to grades and grade points according to the following table.

Marks as Percentage	Grade	Grade Point
[75, 100]	O	6
[65, 74]	A	5
[55, 64]	B	4
[45, 54]	C	3
[35, 44]	D	2
[25, 34]	E	1
[00, 24]	F	0

(B) For courses which have both University Examination (UE) and Internal Assessment (IA):

In order to pass at the University examination in a course, a student must obtain at least 'C' grade. There will be a separate grade assigned for performance in IA. There is no minimum grade for passing in IA. The grade point index (GPI) will be calculated for a course as and when the student passes in the UE by securing at least a C grade. In such a case, the GPI is calculated by combining grade points at UE and IA with respective weights 0.7 and 0.3. A student passes in the course, if the combined GPI is at least 2.5. A student, who passes in the UE but fails in aggregate (this happens only if he/ she obtains only E or F in IA), has to clear the course by re-appearing at the IA only as a back-log candidate. A student, who fails at the UE, has to clear the course by appearing at both UE and IA as a back-log candidate.

(C) For courses which have no University Examination:

There are some courses, the four electives (theory), the lab electives, and the General courses, for which there is no UE. The respective institutes conduct 'continuous assessment' and EoTE. For such courses the computation of GPI, passing criteria, and back-log clearing rules are as in (B) above.

(D) At the end of each semester, a cumulative grade point average (CGPI) will be calculated as a weighted average of the GPI of all courses in which the student has passed till that semester.

(E) A student who passes in all the courses will be declared to have passed the entire MCA programme with the following honours.

CGPI in [2.50, 2.99] -- Pass Class

CGPI in [3.00, 3.49] -- Second Class

CGPI in [3.50, 3.99] -- Higher Second Class

CGPI in [4.00, 4.99] -- First Class

CGPI in [5.00, 6.00] -- Distinction.

(F) PROMOTION RULES:

(i) For admission to Semester V candidate must have passed in all courses of Semesters I and II.

(ii) A student who has cleared all courses of Semesters I and II, is permitted to complete Semester VI before Semester V.

(G) STRUCTURE OF QUESTION PAPERS AT UNIVERSITY EXAMINATION:

Each theory paper is of 100 marks and 3 hour duration. The question paper will have two sections, named Section I and II. Section I is of 60 marks in which the questions tests the Knowledge, comprehension and application skills. There will be six questions, each of 15 marks, of which FOUR are to be answered. Section II is of 40 marks in which the questions test the skills of analysis, creation (synthesis), and evaluation. There will be three questions, each of 20 marks, of which TWO are to be answered.

Each practical examination for compulsory lab courses is of 100 marks and is of three hour duration. There will be three questions of which TWO are to be answered. Each question carries 40 marks and the oral examination in the subject carries 20 marks. The practical examination will be conducted by a panel of examiners appointed by the University and consisting of one internal and one external examiner for a batch of 30 students.

The project courses in Semesters II, III, IV, V, VI will be evaluated out of 100 marks by a panel consisting of one internal and one external examiner for a batch of 30 students and appointed by the University.

9. STRUCTURE OF TRANSCRIPT

At the end of each semester, student will be given a transcript showing the performance and result in each course. The transcript shows, for each course the title of the course, level of the course, credit values, grade in UE, grade in IA, Grade point index, result as pass or fail. Also, the Semester grade point average and cumulative grade point average will be shown. Further, the equivalent percentage of marks corresponding to SGPA and CGPA will be shown. The formula to convert SGPA or CGPA to equivalent percentage is given by:

$$\text{Equivalent percentage marks} = \begin{cases} | 50 \times \text{CGPA} & \text{if CGPI/SGPI is in } [0.00, 0.50] \\ | 10 \times \text{CGPI} + 20 & \text{if CGPI/SGPI is in } [0.50, 5.50] \\ | 50 \times \text{CGPI} - 200 & \text{if CGPI/SGPI is in } [5.50, 6.00]. \end{cases}$$

The above formula corresponds to the following table:

Table: Correspondence between Marks and Grade Point Index (CGPI)

Range of Marks in %	Range of SGPI/CGPI
[75, 100]	[5.50, 6.00]
[65, 74]	[4.50, 5.49]
[55, 64]	[3.50, 4.49]
[45, 54]	[2.50, 3.49]
[35, 44]	[1.50, 2.49]
[25, 34]	[0.50, 1.49]
[00, 24]	[0.00, 0.49]

10. Structure of the MCA Programme:

The MCA programme is of 150 credits. The distribution of credits over the semesters is given below.

Course Type	SEM-I	SEM-II	SEM-III	SEM-IV	SEM-V	SEM-VI	Total Credits	EoTE Univ./ Institute.
Compulsory Theory-Core	3*4 =12C 15 H	3*4 =12C 15 H	3*4 =12C 15 H	3*4 =12C 15 H	2*4 =8C 10 H	-	56 C 70 H	University
Compulsory Theory-Mathematics	1*3 =3 C 4 H	1*3 =3 C 4 H	1*3 =3 C 4 H	1*3 =3 C 4 H	-	-	12 C 16 H	University
Compulsory Theory-Management	1*3 =3 C 4 H	1*3 =3 C 4 H	1*3 =3 C 4 H	-	-	-	9 C 12 H	University
Compulsory LAB Courses	2*4 =8 C 12 H	1*4 =4 C 6 H	1*4 =4 C 6 H	1*4 =4 C 6 H	-	-	20 C 30 H	University
Project Work	-	1*2 = 2 C 4 H	1*2 = 2 C 4 H	1*2 = 2 C 4 H	1*2 =2 C 4 H	-	8 C 16 H	University
Internship	-	-	-	-	-	1*10 =10C	10 C	University
							115 C	
Elective Theory	-	-	-	1*4 =4 C 4 H	3*4 =12C 12 H	-	16 C 16 H	Institute
Elective LAB Courses	-	-	1*3 =4 C 5H	1*3 =4 C 5H	1*3 =4 C 5H	-	3*3= 9 C 15 H	Institute
General Courses	1*2 = 2C 4 H	1*2 = 2C 4 H	1*2 = 2C 4 H	1*2 = 2C 4 H	1*2 = 2C 4 H	-	10 C 10 H	Institute
Total Credits Per Semester	28	26	29	30	30	10	150	
Total Hours Per week	39	37	42	42	40	30	230	

SEMESTER-WISE COURSE INFORMATION

SEMESTER I

Course Number	Course Title	Credit Value	#Lec.	#Tut.	#Lab.	Weightage for EoTE/IA	EoTM
101	Basics of Software Development Skills	4	3	2	-	0.7/0.3	Univ.
102	Elementary Algorithmics	4	3	2	-	0.7/0.3	Univ.
103	Database Management Systems	4	3	2	-	0.7/0.3	Univ.
104	Discrete Structures I	3	2	2	-	0.7/0.3	Univ.
105	Management Functions	3	2	2	-	0.7/0.3	Univ.
106 [#]	Programming with C	4	2	-	4	0.7/0.3	Univ.
107 [#]	Web Supporting Technologies	4	2	-	4	0.7/0.3	Univ.
108	General Course I	2	-	-	4	Continuous Assessment	Institute
Total		28	17	10	12		

SEMESTER II

Course Number	Course Title	Credit Value	#Lec.	#Tut.	#Lab.	Weightage for EoTE/IA	EoTM
201	Computer Architecture	4	3	2	-	0.7/0.3	Univ.
202	Object Oriented Analysis and Design	4	3	2	-	0.7/0.3	Univ.
203	Business Information Systems	4	3	2	-	0.7/0.3	Univ.
204	Discrete Structures II	3	2	2	-	0.7/0.3	Univ.
205	Financial Accounting and Management	3	2	2	-	0.7/0.3	Univ.
206 [#]	Database Management with Oracle	4	2	-	4	0.7/0.3	Univ.
207	Project I	2	-	-	4	0.7/0.3	Univ.
208	General Course II	2	-	-	4	Continuous Assessment	Institute
Total		26	15	10	12		

SEMESTER III

Course Number	Course Title	Credit Value	#Lec.	#Tut.	#Lab.	Weightage for EoTE/IA	EoTM
301	Operating System Concepts	4	3	2	-	0.7/0.3	Univ.
302	Computer Networks	4	3	2	-	0.7/0.3	Univ.
303	Software Engineering	4	3	2	-	0.7/0.3	Univ.
304	Empirical Methods for Researching Information Systems	3	2	2	-	0.7/0.3	Univ.
305	Organizational Behaviour	3	2	2	-	0.7/0.3	Univ.
306 [#]	Programming with Java	4	2	-	4	0.7/0.3	Univ.
307	Project II	2	-	-	4	0.7/0.3	Univ.
308	LAB Elective I	3	1	-	4	Continuous Assessment	Institute
309	General Course III	2	-	-	4	Continuous Assessment	Institute
Total		29	16	10	16		

SEMESTER IV

Course Number	Course Title	Credit Value	#Lec.	#Tut.	#Lab.	Weightage for EoTE/IA	EoTM
CC401	Applied Data Structures	4	3	2	-	0.7/0.3	Univ.
CC402	Design Patterns and HCI	4	3	2	-	0.7/0.3	Univ.
CC403	Probability and Simulation	3	2	2	-	0.7/0.3	Univ.
EC	Elective Course I	4	4	-	-	Continuous Assessment	Institute
EC	Elective Course II	4	4	-	-	Continuous	Institute

						Assessment	
CC406 #	Dot Net Technologies	4	2	-	4	0.7/0.3	Univ.
PR03	Project III	2	-	-	4	0.7/0.3	Univ.
EL02	LAB Elective II	3	1	-	4	Continuou s Assessme nt	Institute
GC04	General Course IV	2	-	-	4	Continuou s Assessme nt	Institute
Total		30	18	08	16		

SEMESTER V

Course Number	Course Title	Credit Value	#Lec .	#Tut.	#Lab .	Weightage for EoTE/IA	EoTM
CC501	Soft Computing	4	3	2	-	0.7/0.3	Univ.
CC502	Finite Automata and Grammars	4	3	2	-	0.7/0.3	Univ.
CC503	Software Project Management	4	3	2	-	0.7/0.3	Univ.
EC03	Elective Course III	4	4	-	-	Continuous Assessment	Institute
EC04	Elective Course IV	4	4	-		Continuous Assessment	Institute
PR04	Project IV	2	-	-	4	0.7/0.3	Univ.
EL03	LAB Elective III	3	1	-	4	Continuous Assessment	Institute
GC05	General Course V	2	-	-	4	Continuous Assessment	Institute
Total		27	18	06	12		

SEMESTER VI

Course Number	Course Title	Credit Value	#Lec.	#Tut.	#Lab.	Weightage for EoTE/IA	EoTM
601	Internship Project	10	-	-	-	0.7/0.3	Univ.

There is no theory examination conducted by university for courses 106, 107, 206, 306, 406.

The four electives, the three Lab electives, and General Courses are finalized by the respective institutes.

MASTER OF COMPUTER APPLICATIONS (MCA) PROGRAMME
Course-wise Syllabus
Semester I

101: Basics of Software Development Skills

(4 Credits, 3L + 2T, Level 3)

Objectives: The main objective of this paper is to emphasize that Systems Thinking and Abstract Thinking are fundamental to efficient and effective software development. So the conventional 'Fundamentals of Information Technology' topics are taught to provide new perspective which is expected to be maintained in all other papers also.

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

- A) Visualize problems as systems and analyze problems carefully,
- B) Have knowledge about common hardware and software systems that help in software development,
- C) Have basic knowledge required for all other courses.

Prerequisites: XII Level Mathematics

Text Book(s): Presently, there is no book which embodies the spirit and simplicity of this course. The faculty from all institutes will jointly prepare unit-wise study material which would be used as a text book.

Syllabus:

Unit 1: System Concepts (3L)

Definition of a System, Elements of system, Characteristics of a System, Types of Systems, Components of TPS, DoS Model, Windows Model, Unix Model, LAN Model.

Unit 2: Hardware System & Data Communication & Networking Concepts :(12L)

Hardware Systems: Block Diagram, Analysis of CPU, Analysis of Internal Memory, Analysis of Arithmetic Logic Unit, Input Devices, Output Devices and Control Unit, Analysis of Firmware.

Data Communication and Networking: Introduction, Data communication Process, Communication concepts, Communication Types, Data transmission Medias, Types of Network: LAN, WAN, MAN, Topologies of LAN, Components of LAN: Media, NIC, NOS, Bridges, HUB, Routers, Brouters, Switch, Repeaters, Gateways, Modem, Communication Protocols, Roles of communication Protocol, layered protocols. Communication Processors: Multiplexers, Concentrators, Front End Processors. Asynchronous & Synchronous Transmission, Switching Techniques. Client/Server Computing.

Unit 3: Software Systems: (6 L)

Operating Systems: Definition of O.S., Functions of Operating Systems, Types and examples of Operating Systems, User Interface of Windows.

Application Software: Word Processors, Spread Sheets, Data Base Management Systems, Presentation Managers, Internet Systems, CAD Systems, Expert Systems

Unit 4: Abstraction: (9L)

Definition and Goals of Abstraction, Major Types of Abstraction- Problem Abstraction, Procedure Abstraction, Data Abstraction, and Program Abstraction. Problem Abstraction: Definition of the terms Problem and Problem Abstraction, Examples of Problem Abstraction.

Procedure Abstraction: Top-Down Approach to Problem Solving, Definition of Procedure and Procedure Abstraction, Examples. Data Abstraction: Nature and Types of Data, Internal and External Representations of Data, Bits and Bytes, Examples. Representation of Non-numeric Data- ASCII, Unicode. Examples. Program Abstraction: Definition of Program Abstraction and concept of an algorithm, Algorithm Representation-Pseudo Code, Algorithm Representation- Flow Charts, Elementary Algorithms for Search, Sort, and Merge Sort. Recursive Algorithms: Definition and Examples.

Unit 5: Representations :(12L)

Number Representation: Representation of Non-negative Integers, radix-r representation, conversion from radix-r to radix-s, Addition and Subtraction of Radix-r Integers, Error Detection, Multiplication of Radix-r Integers, Division of Radix-r Integers, Computing the remainder, BCD Representation and Arithmetic. Representation of Integers: Sign-Magnitude forms and r-1 and r complement forms, Arithmetic in r complement form. Representation of Rationals: Definition of Rational Numbers, Rational Arithmetic; Representation of Real Numbers: Mantissa and Fraction form, Exponent form, Normalization, IEEE 754 form, concept of precision, Addition and Subtraction of Floats, Multiplication and Division of Floats, Errors in Floating Point Arithmetic. Representation of Strings: Definition of String, Counted String Representation, Null Terminated Strings, String Manipulation.

Unit 6: Computer Language :(6L)

Introduction to Computer Languages, Language Types, Definition of Program, Programs in Machine Language, Assembly language and Programs in Assembly Language, Interpreters, Higher Level Languages, Major Paradigms-Procedure Oriented, Object Based, Object Oriented, Logic Based, Functional languages, Examples. Compilers and Translation Process

Programming: Meaning of Programming, Concept of Modular Programming.

Unit 7: Computer Maintenance, privacy & Security :(3)

Computer Maintenance- Introduction, Typical causes of System Failure & Disaster Recovery.

Computer Virus Computer Viruses, Trojan horse, Worms. Some reported viruses. File infectors, Disk Infectors, Partition Infectors, Boot Infectors, Companion Viruses, CMOS Virus, and Hardware Virus. Impact of Information Technology on Society, Computer Privacy, Security and Integrity.

102: Elementary Algorithmics

(4 Credits, 3L + 2T, Level 3)

Objectives: This is a foundational course. The course aims at developing the logical abilities of the student. The student will be exposed to varieties of problems and problem solving strategies.

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

- A) State precisely the problems and analyze the problems following the top-down approach;
- B) Design algorithms to standard problems following the Brute Force, Divide and Conquer, Decrease and Conquer, and Transform and Conquer strategies;
- C) Analyze the algorithms in terms of time complexity

Pre-requisites: XII standard Mathematics

Text Books:

1. Levitin A. (2007), Introduction to the Design and Analysis of Algorithms, Second Edition, Pearson Education.
2. Dromey R. G. (1982), How to Solve It by Computer, Prentice Hall India

Syllabus:

Unit 1: Concepts of a problem and instances of a problem, Solution of a problem, Procedures and algorithms, algorithm representation, Pseudo code language and Flow Chart language, Tracing of algorithms, Standard problem types – Problems of Sorting, Searching, and Merging, Problems of String Processing, Graph Problems, Combinatorial Problems, Geometric Problems, and Numerical Problems. (6L)

Unit 2: Concepts of Procedure Oriented Programming- Character set, Tokens, Key words and Identifiers, Constants, Statement types, Arithmetic and logical expressions, Syntax, semantic, and logical errors, data types, structured programming- sequence, selection, and iteration structures; one and two dimensional arrays, records, functions and procedures, parameter types, local and global variables, files and file concepts. (12L)

Unit 3: Factors that affect the performance of an algorithm, time and space complexities, Need for models of Computing, Operation Counts, Asymptotic Complexity measures.(6L)

Unit 4: Algorithm Design Methods- Brute Force Method, Divide and Conquer Method, Decrease and Conquer Method, Transform and Conquer Method. Simple applications of the methods to various standard problems. (15L)

Unit 5: Recursive algorithms and relation to iterative algorithms, execution of recursive algorithms, Analysis of recursive algorithms. Applications to standard problems. (6L)

103: Database Management Systems

(4 Credits, 3L + 2T, Level 3)

Objectives: This is a foundational course on Data Modeling. The course aims to impart knowledge of the concepts related to database and operations on databases. It also gives the idea how database is managed in various environments with emphasis on security measures as implemented in database management systems.

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

- A) Understand the concepts of database and techniques for its management.
- B) Different Data Models at Conceptual and Logical level.
- C) Differentiate between the role of DBA and Data Architect
- D) Understanding Data Security standards and Methods

Pre-requisites: XII Standard Mathematics

Text Books:

Korth.(2006) Database Systems Concepts , Tata McGra-Hill, Fifth Edition.

Syllabus:

Unit 1: Basic Concepts and Architecture – (3L)

Database and Need for DBMS, Database System versus File Systems, Database Users, 3-tier Architecture, View of Data, Instances and Schema.

Unit 2: Data Modeling – (9L)

Logical Data Modeling, Hierarchical, Network , Relational, Advantages and Disadvantages, Conceptual Data Modeling –Entity –Relationship Model Entities, Relationships, Keys, different types of attributes, E-R Diagram conventions, degree, cardinality, participation, composite Entities, Strong and Weak Entity, E-R Diagram with case study, Extended E-R and Relational Model, Superclass and Subclass Entity types, Attribute inheritance Aggregation, Specialization, Generalization with examples, Logical view of Data, (Table and its characteristics) Concept Domain, tuple. Object Oriented data Modeling-Object and Object Identifier, Attributes and Methods, Class, Object Structure, Object Classes with example. Inheritance, Multiple inheritances, Object Identity, Object containment.

Unit 3: Relational Algebra – (6L)

Keys (composite, candidate, primary, secondary) Foreign, Integrity Rules, Relational Set Operators -Union, Intersect, Difference, Product, Select, Project, ,Divide, Assignment. Set Operators Join, Outer Join, Inner Join, with example, Relationship within the Relational Database(1:M,1:1,M:N) with example. Problems and Reduction of M:N with example. Codd's Rules Examples of RDBMS that implements some Codd's rules.

Relational Database Design, Mapping Conceptual Model into relational model with example/case study

Unit 4: Normalization – (6L)

Functional Dependencies, Normal Forms (1NF, 2NF, 3NF, BCNF) with examples. Case Study on normalization, Decomposition, Lossless join and dependency preserving Decomposition.

Unit 5: File Structure and Data Administration – (6L)

File Organization, Organization of records in file (Sequential, Clustering), Indexing and Hashing. Basic concepts, indices, B+ Tree index file, B- tree index file, Static Hashing, dynamic Hashing. Data administration, Role and responsibility of DBA, creating / deleting / updating table space, database monitoring, user management.

Unit 6: Transaction and Concurrency Control – (9L)

Concept of transaction, transaction state, ACID properties, Serializability, Lock based protocols, timestamp based protocols, Multiple Granularity, Multiple Version Techniques, Deadlock and its Handling. Backup and Recovery Techniques, Failure Classification, Recovery & Atomicity, Log Based Recovery, Checkpoints, Shadow Paging, Recovery with concurrent transactions, Failure with loss of non-volatile storage.

Unit 7: Data Quality and security management – (6L)

Basic data security principles – user privileges, data masking, encryption and decryption. Data Security Implementation, ITCS304 (security standard) revalidation of user, role, privileges. Data Quality Management, Basic quality principles, data quality audit, data quality improvement.

104: Discrete Structures-I

(3 Credit, 2L+ 2T, Level 3)

Objectives:

This course provides the mathematical foundations for many computer science courses including data structures, algorithms, database theory, automata theory, compiler construction and operating system.

Learning Outcomes: At the end of the course student should be able to:

- A) Solve Mathematical reasoning
- B) Developing the problem solving skills

Pre-requisites: XII Level Mathematics

Text Book(s)

Kenneth H. Rosen , Discrete Mathematics and its Applications Edition 6th Tata McGraw Hill

Syllabus:

Unit -1: Propositional and Predicate Logic- (15L)

Truth tables, Connectives, Conditional statements, ,Tautologies and fallacies, Logical expressions and their equivalence, Disjunctive and Conjunctive normal forms, satisfiability of a proposition. Predicates and quantifiers, Logical equivalence involving quantifiers, Negating quantified expressions, introduction to logical programming, Nested quantifiers, order of quantifiers, Rules of Inference for propositional logic, Building arguments, Principle of Resolution, Rules of Inference for predicate logic. Skolmization ,Introduction to Proofs: Basic Terminology, Direct Proofs, Proof by Contradiction, Proof methods and strategies

Unit-2: Sets- (6L)

Set operations and their properties, Power set, partitions, Cartesian products ,Computer representation of sets, Functions- One-to-one and Onto functions, Pigeon hole principles, inverse functions and composition of functions, Cardinality of sets, countable and uncountable sets ,cantor diagonalization argument, permutations and their properties.

Unit -3: Introduction and Recursion- (9L)

Mathematical induction and its applications, strong form of mathematical induction, Generalized Induction and its applications, Application of Mathematical induction to establish Programme Correctness, Recursive definitions, Lame's Theorem.

Unit-4 Relations-(6L)

Basic, concepts, functions as relations, Reflexive, symmetric, asymmetric, Anti-symmetric, transitive, equivalence, partially ordered relations. Algorithm to verify the simple properties of relations, Combining relations-set operations, inverse and compositions of relation, N-ary relations and their applications.

Unit-5 Representing Relations-(9L)

Boolean matrices and operations on Boolean matrices, Corresponding between Relations and Boolean matrices, Basic concepts of graph and directed graphs, Representing of relations as directed graphs, Closures of Relations, Warshall's algorithm, Equivalence classes induced by a relation

105: Management Functions

(3 Credits, 2L+2T, Level 2)

Objectives:

1. To orient the students to principles of management
2. To make them comprehend the process of management
3. To internalize the principles through rigorous assignments where they shall observe, analyze and infer the presence of principles transformed into practice.

Learning Outcome:

At the end of the course, the students shall acquire

1. Understanding of functions of management
2. Understand the principle of management woven in to the process of management
3. Understand how they are modified in to practice to suit the requirements
4. How IT influences the process of management

Prerequisites : Inquiring Mind and Strong inclination for observation

Reference Books:

1. H.Welrcih, Mark Cannice, H. Koontz, Management , A Global and Entrepreneurial Perspective , McGraw-Hill Companies, 12th edition.
2. P.C.Tripathi, P.N.Reddy, Principles and Practice of Management , Tata Mcgraw Hill , 3 ed.
3. L.M. Prasad, Principles and Practice of Management, Seventh Edition
4. Stephan Robbins, Mary Coutler, Management

Syllabus

Unit I : (05 L)

The need of Management Study , Process of Management , Characteristics of Professional Management , Brief Review of Management Thought Social Responsibility of Management

Unit II: (04 L)

Decision Making Process , Planning and Steps in Planning , Types of Plan Making Planning Effective , Case Study on Planning, MBO

Unit III : (06 L)

Organization, Meaning and Process , Departmentalization,, Organization Structure , Authority and Delegation , Centralization verses Decentralization , Team Work , Case Study

Unit IV : (03 L)

Co-ordination – meaning and need , Techniques of establishing Co-ordination
Difficulties in establishing co-ordination , Case Study

Unit V : (05 L)

Formal and Informal Organization, Manpower Planning , Recruitment and Performance Appraisal, Compensation and Incentives , issues related to Retention
Case study

Unit VI : (03 L)

An overview of Communication, Supervision and Direction , Leadership Styles , Control – need and types and control techniques .

In addition there shall be tutorials of written examination type, field study and presentation.

106: Programming with C

(4 Credits, 2L + 4P, Level 3)

Objectives : This is a foundational course in programming. The course aims at practically developing the logical abilities of the student using C language. The student will be exposed to programming language C. Emphasis is on semantics and problem solving.

Learning Outcomes :

At the end of the course, student should be able to efficiently solve the problems and analyze the problems using of C language.

Pre-requisites : XII standard Mathematics

Text Books :

- 1.E. Balagurusamy (2009), Programming with C , Tata McGraw Hill
2. Yashawant Kanetkar, let Us C, BPB Publication

Syllabus :

Unit-I : (2L)

Origin of C, History, Structure of C Program, Characters and Character Set of C, Tokens in C- Identifiers, Keywords, Constants, Operators(arithmetic, relational, logical, assignment, bitwise, conditional, other operators), precedence and associativity rules. Formatted input and output , Data Types,

Unit-II : (5L)

Executable and Non-Executable Statements, Types of Executable Statements- Input Statements, Assignment and Arithmetic Statements, Control Statements- Sequential, Selection(if, if--else, switch..case), Iteration Statements(do..while, while, for), Output Statements.

Unit-III : (6L)

Functions in C : Introduction, Advantages, Standard library functions, User defined functions : Declaration, definition, function call, parameter passing (by value), return keyword, void, Scope of variables, storage classes, Recursion, Arrays : Declaration, initialization, Types – one, two and multidimensional, Passing arrays to functions

Unit-IV : (4L)

Pointers : Direct and Indirect Access; need of pointers, de-referencing, constant and variable pointers. Semantics of array and function declarations. Using key word const in array declarations, Function pointers. pointers as arguments to functions and as return types, generic pointer void *, Parsing pointer declarations,
Dynamic memory allocation : malloc, calloc, realloc and free

Unit-V : (2L)

Strings : Declaration and initialization, Standard library functions, Strings and pointers, Array of strings,

Unit-VI : (4L)

Structures and Unions : Creating structures, Accessing structure members (dot Operator), Array of structures, Pointers and structures, Unions, Difference between structures and unions, enum declaration

Unit-VII : (2L)

C Preprocessor : Format of Preprocessor directive, File Inclusion directive, Macro, Command Line Arguments : Accessing command line arguments

Unit-VIII : (5L)

File Handling : File organization (Sequential, Direct Access, Indexed Sequential), Operations on files, Random access to files

References :

1. Kernighan and Ritchie : The C Programming Language
2. Ellis Horowitz, Sartaj Sahni : Fundamentals of Computer Algorithms (Galgotia)

Laboratory Experiments in C

1. Program to display whether inputted number is Armstrong number or not.
2. Program to display all numbers divisible by 7 between 1 to 100.
3. Program to input any number and display digits which are absent in given number.
4. Program to display all prime numbers between 1 to 500.
5. Program to display sum of digits of a given numbers upto a single digit.
6. Program to input any five digit number and display its last and first digit.
7. Program to convert rupees into thousands, hundreds and rupees
8. Display following pyramid

(i)	1 1 1 1 1	ii)	1
	2 2 2 2		2 1 2
	3 3 3 3		3 2 1 2 3
	4 4 4		4 3 2 1 2 3 4
	5 5		
	6		
9. Program to calculate LCM and GCD of two numbers.
10. Program to calculate prime factors of a given number.
11. Program to input any decimal number and convert it into various bases.
12. Program to check whether inputted number is palindrome or not.
13. Program to calculate $1/1!+2/2!+\dots+n/n!$
14. Read a line of characters from the user and count no of lines, words, spaces, tabs and characters in it.
15. If n is any inputted numbers then determine whether n is the sum of all of its divisors i.e n is sum of all t such that $1 \leq t < n$, and t divides n.
16. Program to calculate nCm .
17. Program to display GCD using recursion.
18. Program to calculate Fibonacci series using recursion.

19. Program to calculate $n!$ using recursion.
20. If a is an array of n elements then write recursive function to display powerset of a .
21. Program to display all array elements in ascending order using selection sort.
22. Program to display all array elements in descending order using bubble sort.
23. Program to calculate largest and second largest from a set of n numbers.
24. Program to perform various operations on matrix (addition, subtraction, multiplication, norm of matrix, saddle point, magic square, inverse, transpose)
25. Program to display students list in ascending order.
26. Program to perform various string operations (find length, checking palindrome, copy one into another, comparing strings, find substring using user defined functions)
27. Display marksheet of student using structure.
28. Program to display bazaar bill of any customer.
29. Program to copy contents of one file into another.
30. Program to display contents of a file in uppercase letters.

107: Web Supporting Technologies

(4 Credits, 2L + 4P, Level 3)

Objectives: - To impart the basic knowledge and skills required to develop web applications using HTML, Cascading Style Sheets and JavaScript.

Learning Outcomes:- Upon successfully completing this course the student will be able to

- Apply the concepts and the principles of WWW
- Should be able to use CSS and JavaScript to achieve DHTML effects.

Text Book:- Ivan Bayross (2006) Web Enabled Commercial Application Development Using HTML, DHTML, JavaScript, Perl CGI, BPB Publications.

Syllabus :-

Unit 1:- Overview of Internet and Intranet (2L)

Understanding internet and its need, concept of intranet, difference between internet and intranet, a brief history, internet applications, Internet Service Providers (ISP). Concept of client and server, web browser and web server, communicating on the internet, concept of domain- Physical domain, virtual domain, registering a domain. Need of IP addressing, Process to assign IP addresses, World Wide Web

Unit 2:- Introduction To HTML(2L)

Overview of HTML, concept of Tag, types of HTML tags structure of HTML program with simple examples. Text Formatting Through HTML: Emphasizing Material In a Web Page: Using Image tag, attributes of Image tag, Using Lists: unordered, ordered, definition lists.

Unit 3:- Handling Tables, Frames and Linking Documents (5L)

Tables:- To define header rows & data rows, use of caption tag, changing height & width of table, cellpadding, cellspacing, bgcolor, colspan, rowspan.

Hyperlink:- Concept of hyperlink, types of hyperlinks (Internal & External), linking to the beginning of document, linking to a particular location in a document, image as hyperlinks and image mapping.

Frames:- Introduction to frames with its attributes, using frames & framesets, named frames.

Unit 4:- Cascading Style Sheets (4L)

Introducing CSS, font attributes, color and background attributes, text attributes, border attributes, margin related attributes, list attributes. Using class and span tag

Types of style sheets: inline, embedded and external Style Sheets.

Unit 5:- Introduction To JavaScript (2L)

Introduction to scripting, overview of Java Script, advantages, client side java Script, capturing user input, writing Javascript into HTML

Data types, literals, variables and operators, Java Script arrays, dense array, operators, expressions

Unit 6:- Java Script Programming Constructs (4L)

Assignment, data declaration, if, switch, while, for, do while, label, break, continue, function call, return, with, delete, method invocation.

Types of functions in Java Script:- Built in functions, User defined functions, function declaration, passing parameters, variable scope, return values, recursive functions. Placing text into browser, window objects.

Dialog boxes - Alert dialog box, prompt dialog box, confirm dialog box.

Unit 7:- Java Script Document Object Model (3L)

Understanding JDOM, Java Script Assisted Style Sheets, understanding objects in HTML Browser objects, how a javascript enabled browser handles the document object, Form element array. Access to elements of a web page, manipulation of web page element. Handling web page events using Javascript, Javascript event handlers.

Unit 8:- Forms Used By Web Site (8L)

Form object, form object's Method, properties of form elements, methods of form element, Different elements - text, password, button, submit, reset, checkbox, Radio, TextArea, select & option.

Other built-in Object- String object, math object, date object, User defined objects- creation, instances, objects within objects.

Introduction to ASP, ASP request Processing, generating response, using session and cookies, application object, database connectivity and role of global.asa file.

108: General Course I (Communication skills)

(2 Credits)

Objectives: A widely-heard theme among employers is that IT professionals must be able to communicate effectively with colleagues and clients. Because of the importance of good communication skills in all computing careers, MCA students must sharpen their oral and writing skills in a variety of contexts -- both inside and outside of IT courses. In particular, students in IT programs should be able to:

- Communicate ideas effectively in written form
- Make effective oral presentations, both formally and informally
- Understand and offer constructive critiques of the presentations of others
- Have a pleasant demeanor as they work with people on their IT needs, either in person or by phone
- Write appropriate electronic communications (including email, blogs, instant messages, etc.) to all levels of workers in all IT endeavors.

While institutions may adopt different strategies to accomplish these goals, the program for each IT student must include numerous occasions for improving writing and practicing oral communication in a way that emphasizes both speaking and active listening skills.

At a minimum, an IT curriculum should require:

- Course work that emphasizes the mechanics and process of writing
- At least two formal oral presentations to a group
- The opportunity to critique at least two oral presentations

The institutes are expected to devise the course meeting the above objectives, keeping in mind the requirements of their students.

MASTER OF COMPUTER APPLICATIONS (MCA) PROGRAMME
Course-wise Syllabus

Semester II

201: Computer Architecture

(4 Credits, 3L+2T, Level 3)

Objective: Main objective of this paper is to learn structure and functioning of various hardware components of digital computer. Also study the interactions and communication among these hardware components.

Learning Outcomes:

At the end of this course, student should be able to understand

- Simple machine architecture and the reduced instruction set computers.
- Memory control, direct memory access, interrupts, and memory organization
- Basic data flow through the CPU (interfacing, bus control logic, and internal communications).
- Number systems, instruction sets, addressing modes, and data/instruction formats.

Text Book(s): M Morris Mano Computer systems Architecture third edition Prentice Hall of India Publication

UNIT 1.Introduction To Digital Computer – (7L)

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

Unit 2.Introduction To Digital Components And Micro Operations – (6L)

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.

Unit 3.Computerorganization And Programming – (8L)

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.

Unit 4: Memory Organization And CPU – (8L)

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, RISC

Unit5: Pipeline And Vector Processing – (5L)

Parallel processing – Pipelining - Arithmetic pipeline - Instruction pipeline - RISC pipeline, - Vector processing - Array processor

Unit6: Input – Output Organization – (5L)

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

Unit7: Multiprocessors – (6L)

Characteristics of Multiprocessors – interconnection structures – interprocessor Arbitration – interprocessor communication and synchronization – Cache coherence

202: Object Oriented Analysis and Design

(4 Credits, 3L + 2T, Level 4)

Objectives: The course aims at developing skills to analyze and design a software system using Object Oriented Analysis and Design (OOAD) and UML. And use these skills in Unified Process (UP) environment.

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

- A) Understand and describe the Object Oriented concepts
- B) Describe Object Oriented Analysis and Design(OOAD) concepts and apply them to solve problems
- C) Prepare Object Oriented Analysis and Design documents for a given problem using Unified Modeling Language
- D) Describe the activity carried out in each and every phase of Rational Unified Process(RUP)

Text Book:

1. Craig Larman (2004), Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd Edition, Pearson Education.

Reference Books:

1. Martin Fowler (2003), UML Distilled, 3rd Edition, Pearson Education.
2. Roger Pressman(2009), Software Engineering: A Practitioner's Approach, Roger Pressman, ; 7th edition, McGraw-Hill
3. Brett D. McLaughlin (2006), Head First Object-Oriented Analysis and Design , 1 edition, O'Reilly

Syllabus

Unit 1: (6L)

Introduction to OO Concepts, Object, Abstraction, Class, Encapsulation, Message Sending, Inheritance, Polymorphism, Association, Aggregation, Concept of analysis and design, introduction to Object Oriented an Analysis and Design(OOAD), review of Software Development Life Cycle(SDLC), Iterative development and Rational Unified Process(RUP), Phases of RUP

Unit 2: (6L)

Understanding Inception and requirement, writing requirements in context, use case modeling, relating use cases to user goals and elementary business process, use cases for adding values and functional requirement, actors, actor types and goals in use cases, writing use cases in UI-free style, moving from inception to elaboration.

Unit 3: (6L)

Elaboration Iteration 1 emphasize and requirement, System behavior, System Sequence Diagrams, Inter-System SSDs, SSDs and Use Cases, System Events and the System Boundary, Naming System Events and Operation, SSD with Use Case Text, SSDs within the UP, UP Artifacts

Unit 4: (6L)

Domain Modeling, Conceptual Class Identification, Resolving Similar Conceptual Classes-Register vs. "POST", Specification or Description Conceptual Classes, UML Notation, Models, and Methods: Multiple Perspectives Lowering the Representational Gap, Domain Models Within the UP, Finding and adding association, Common Associations List and Guidelines, Association Roles, Naming Associations, finding attribute and its types, UML Attribute Notation, attributes and foreign Keys, Multiplicity

Unit 5 : (6L)

Use cases and operation contracts, pre and post conditions, contracts leads to domain model, contracts usefulness and guidelines, contract operations and UML, contracts within UP, GRASP: Designing Objects with Responsibilities, Patterns of General Principles in Assigning Responsibilities, Object Design and CRC Cards, Use Case Realization with GRASP Patterns, Determining visibility, Design Class Diagrams(DCD), from elaboration iteration 1 to 2, Iteration 2 emphasize and requirements, Refinement of Analysis-oriented Artifacts in this Iteration 2.

Unit- 6: (7L)

Elaboration Iteration 3 requirements and emphasis, Use case relationships (include, extend and generalize); Concrete, Abstract, Base, and Addition Use Cases, Modeling generalization, conceptual superclass and subclass, abstract conceptual class, modeling changing states, Class Hierarchies and Inheritance in Software, Association Classes, aggregation and composition, Association Role Names, Roles as Concepts vs. Roles in Associations, Derived Elements, Qualified and Reflexive Associations, Ordered Elements, Using Packages to Organize the Domain Model

Unit 7: (8L)

Modeling behavior in statechart diagram, events, states, and transitions in statechart Diagrams, statechart diagrams in the UP, Use Case Statechart Diagrams, classes that benefit from statechart diagrams, external and Interval events, additional statechart diagram notation, Activity diagrams, activity diagrams VsStatechart diagrams, concept of interface, interface realization, using stereotypes in UML, collaboration diagrams , putting sequence diagram as collaboration diagrams, component and deployment diagrams.

203: Business Information Systems

(4 Credits, 3L + 2T, Level 4)

Objectives : The main objective is to

- a) Acquaint students with basic concepts and major issues of Business Information Systems
- b) To describe current techniques and tools used for analysis and design.
- c) To understand and develop the functionalities of various types of systems.

Learning Outcomes :

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

- (a) Analyze user requirements using different Information gathering tools.
- (b) Apply systematic approaches to software development
- (c) Provide solutions to different systems.

Pre-requisites: Students should know basic concepts of computer systems and DBMS.

Text Book(s)

- 1) Elias M. Awad Systems Analysis and Design
- 2) V. Rajaraman Analysis and Design of Information Systems
- 3) Dr. Arpita Gopal Engineering MIS for strategic Business Processes
- 4) James A. Senn Analysis & Design of Information Systems
- 5) S. Parthasarthy and B.W. Khalkar System Analysis & Design
- 6) S.A. Kelkar Structured System Analysis and Design

Unit 1 : Introduction to system concepts(3L)

Introduction to system Definition of system, its characteristics. Elements of system and system types. Categories of system. Examples of system: - Business, Computer, Human as system. Computer based information systems (MIS, DSS, ES) Phases of SDLC: - a) Preliminary Investigation (Problem Identification Feasibility Study with its types) b) System Analysis. c) System Design. d) Coding. e) Testing. f) Implementation. g) Maintenance.

Unit 2 : System Development Life Cycle (6L)

Various models of SDLC:- Waterfall, Rapid Application Development model, Prototyping model, Spiral model, Qualities of Software, Introduction to Agile Methodology.

Unit 3 : Information Gathering Techniques(3L)

Record Review about firm, user staff, work flow from various literatures, forms, Manuals, On site Observations. Interviews and its types, Questionnaires.

Unit 4 : Structured Analysis Tools(6L)

- a) Data Flow Diagram b) Data Dictionary, c) Structured English, d) Decision Trees
- e) Decision Table and its types.

Unit 5 : Designing Methodologies(6L)

Designing process: - Logical and Physical. Designing Tools: Structured Charts, Functional Decomposition (concept of module with coupling and cohesion), System Flowcharts, Entity Relationship Diagram. Normalizing Relations: First Normal form, Second Normal form, Third Normal form, Boyce-Codd Normal Form(BCNF). Data Input Methods: Coding techniques, Detection of Error in code, Validations, Interactive Data Inputs. Output Design : Objectives of Output Design, Design of Output Reports, Design of Screens.

Unit 6 : Control, Audit and security Control in Information system, Audit of Information System, Testing of Information System, Security of Information System. (3L)

Unit 7: Overview of different Business Application Systems(18L)

Education Institute Management System, Library Management Systems, Inventory Management System, Bank Management System, Hospital Management System, Payroll System, Financial Accounting System, Hotel Management System, Human Resource System.

204: Discrete Structures II

(3 Credits, 2L+2T, Level 3)

Objectives:

The aims of this Graph theory is a delightful playground for the exploration of proof techniques in discrete mathematics and its results have applications in many areas of the computing, social and natural science,

Learning Outcomes:

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

- A) Use graphs as models in a variety of areas.
- B) Formulate several real- world problems in mathematical terms

Pre-requisites: Discrete Structures I

Text Book:

Kenneth H. Rosen , Discrete Mathematics and its Applications Edition 6th Tata McGraw Hill

Syllabus:

Unit -1(6L)

Preliminary concept, Introduction, Konigsberg Bridges problem, Utilities problem, seating arrangement problem

Concept of Graph, Graph models, types of Graph, representing Graph, incidence and Adjacency Graph, Graph Isomorphism, Graph representing in computer, planner and Dual Graph

Unit -2 (12 L)

Walk, paths and Euler circuit, Hamiltonian Graph and Graph Algorithm, Konigsberge Bridge problem, city route ,puzzle problem, seating arrangement problem, Travelling salesman problem ,Warshall's algorithm, Floyde's algorithm, Dijkstra's algorithm, Depth-First Search in Directed Graph, coloring Graph and their theorem, Applications of Graph coloring

Unit -3 (12 L)

Concept of Trees, definition and properties of Trees, Application of Trees, Trees as Models, Game Trees, Tree Traversal, Infix and Postfix notation, Binary Trees and Properties , Spanning Tree, Minimum spanning Tree, Depth First search, Breadth –First search, Back tracking applications, Kruskal algorithm, Prims algorithm, Huffman's algorithm, sorting technique (Merge sort, Heap sort),computer representation of Tree

Unit-4 (9L)

Basic counting principles-The product and Sums rules, The Inclusion-Exclusion principle, The Pigeonhole principle, Generalized Pigeonhole principle ,Application, Permutation and Combination, Generalized Permutation and Combination, Generating Permutation and Combinations, Permutation with Indistinguishable objects, Binomial coefficients and their properties, Algorithm to compute Binomial coefficient Pascal Identity and triangle

Unit-5 (6L)

Advance Counting Technique-Recurrence relations and modeling with recurrence relations, Solutions to Linear recurrence relations, Divide and Conquer algorithms and recurrence relations, Master Theorem regarding recurrence relations

205: Financial Accounting and Management

(3 Credits, 3L+1T, Level 2)

Objectives:

1. To orient the students to basic concepts of accounting. costs
2. To make them understand the techniques of management accounting
3. To make them understand the concepts of financial management
4. To give the dimensions of three aspects finance application background in IT packages

Learning outcome:

At the end of the course the students shall be able to

1. The student shall be able to have basic understanding necessary for development of software for accounting and cost and finance
2. Interrelation of concepts as dimensions of financial management process.

Prerequisites: Good understanding of nature of business expenses and income/ revenue

Recommended Books:

1. Ashok Sehagal and Deepak Sehagal , Financial Accounting – Taxman
2. Tulsian – Financial Accounting
3. Jain Naramg – Cost Accounting Principles and Practice
4. I.M. Pandey ,FinacialMangementVikas Publishing House
5. Book Keeping and Accountancy, M.G. Patkar

Syllabus:

Unit 1 : Introduction to Financial Accounting(10 L)

Principles of double entry book keeping, preparation of journal , ledger, trial balance and final accounts (Trading and Profit and Loss Account and Balance Sheet) for a sole proprietor. Accounting concepts and conventions

Unit 2 : Introduction to Cost and Techniques (06 L)

Concept of cost , classification of cost and preparation of cost sheet , Budgetary Control – concept and importance, Simple problems on flexible and cash budget

Unit 3 : Standard and Marginal Costing (08 L)

Concept of Standard Cost , Variance , advantages and limitations , Computation of basic variances of material and labour cost , concept of marginal cost P/V ration . Contribution, BEP, Margin of Safety and problems on above

Unit 4 : Introduction to Financial Management : (04 L)

Scope and Nature of Financial Management, Consideration while preparing financial plan.

Concept of Profit Maximization and Wealth Maximization .

Advantages and Limitation of Long Term Sources of Finance, Venture Capital

Unit 5 : Sources of Short Term Finance (07 L)

Concept of inventory management and EOQ.

Concept of Working Capital and factors affecting working capital, simple problems on estimation of working capital.

Unit 6 : Investment Decision(10L)

Concept of Time value of money, Risk and Return, Simple problems on Capital Budgeting Techniques – Simple Pay Back , Discounted Pay Back , NPV, ARR, Profitability Index

Ratio Analysis : (At theory level) significance of ratios with reference to examples of leverage ,profitability , activity, liquidity ratio

206: Database Management with Oracle

(4 Credits, 2L + 4P, Level 3)

Objectives: The main objective is to teach the concepts related to database its techniques and operations. SQL (Structured Query Language) is introduced in this subject. This helps creates strong foundation for application of data design.

Learning Outcomes: At the end of this course, student should be able to (a) Simple Query using sample datasets(b) Complex queries using SQL.(c) Writing PL/SQL blocks (d) Introducing Object Oriented Concepts in Oracle.

Pre-requisites: Concept of Database Management Systems

Text Books:

1. Ivan Bayross SQL, PL/SQL The Programming Language of Oracle 3rd Revised ed.
BPB Publications

Syllabus:

Unit 1: Introduction (6L)

Concepts of RDMS, Introduction to Oracle, SQL, Defining a database in SQL, creating table, Dual, Arithmetic Operators, comparison operator, Logical operator,

DDL Commands – Creating table, changing table definition, removing table, Creating Tables with constraints on row level and column level, primary key, foreign key, check ,alter, copy. creating data integrity controls, DML Commands- Inserting, updating, deleting data – batch input, deleting database content, changing database content, processing single table clauses of select statement, using expressions, using functions, using wildcards, sorting results; order by clause, categorizing result; group by Having clause. Project Case – Library, Hospital, Reservation system. SET Operations, Intersect, Union, Minus, Union ALL, SUBMULTISET, MULTISET UNION. DCL commands- Granting and Revoking Permissions, Revoking Privileges given

Unit 2: Database Objects: (3L)

Creating index, simple index, composite index, unique index, dropping indexes, multiple indexes on table, using rowed to delete duplicate rows from a table, Creating sequence, altering sequence, dropping sequence, Views -Using and defining, modifying, deleting. Insert / Drop / filter view. Complex view.

Unit 3: Joins and Oracle Functions: (3L)

Table Alias, Inner Join, Outer Join, Self Join, cross join, Outer join left right, Aggregate Functions , Numeric Functions , String Functions, Conversion functions, Date conversion functions, Date functions

Unit4: Introduction to PL/SQL programming – (6L)

What is PL/SQL, Structure of PL/SQL block, header section, declaration section, execution section, exception section. Creating a simple PL/SQL procedure, procedure versus functions, creating stored procedure, creating a functions , PL/SQL variable and constants. Declaring PL/SQL variables and constants, assigning values to variable. Using variables, control structures in PL/SQL – conditional control, iterative control, sequential control. Error handling – Exceptions, System defining exceptions, user defined exception with example.

Unit 5: Cursor and Triggers – (9L)

Types of cursor, implicit cursor attributes, explicit cursor attributes, functionality of Open ,fetch and close commands , cursor FOR loops, parameterized cursors. Triggers- Introduction, Database Triggers, Database Triggerv/s Procedures,Database Triggers v/s Declarative Integrity constraints, Types of Triggers –Row Triggers, Statement Triggers, Before Triggers, After Triggers, Combination Triggers, Deleting a Trigger

Unit 6: Object oriented concepts –(3L)

Introductions to Object Oriented concepts in Oracle. V-array, object type, nested table with examples.

Assignment on PL/SQL

- 1) Print the name, salary and deptno of all employees who belong to deptno 10.
- 2) Print the name, job of employees having job as manager or analyst.
- 3) Write a PL/SQL block to print 3rd, 6th, and 7th record from employee.
- 4) Write a PL/SQL block which assign comm = 500 for those employee who are getting null commission.
- 5) Update the salary of employee by 20 for even records and 100 for odd records.
- 6) Print the information of all employees using cursor.
- 7) Print 4th, 6th and 10th records from employee table.
- 8) Print the names of employee having commission as null.
- 9) Print the information of 1st five highest salary earner.
- 10) Update salary of dept 10 by 1000 and dept 20 by 2000 using cursor.
- 11) Accept deptno and print the number of employees working in that department.
- 12) Accept the deptno and print department name and location.
- 13) Accept employee name and print date of joined.
- 14) Print the name of employee having maximum and minimum salary.
- 15) Accept salary and print number of employees having salary less than or equal to accepted salary.
- 16) Print the name of employees working in department no 10 and having maximum salary.
- 17) Print the number of employees joined in the month of December.
- 18) Increment the salary of employee by 1000 having location as New York.
- 19) Increment the salary of employee 'Raj' by 10% and of 'Ravi' by 20%.
- 20) Check whether Smith's salary is greater than Black's salary or not.
- 21) If Smith's salary is greater than Black's salary then update employee table and set Black's salary same as Smith's salary otherwise set Smith's salary same as Black's salary.
- 22) Write PL/SQL block to process an order for keyboard.
 - a. Check Quantity > 0
 - b. If Yes then insert in purchase order 'One Keyboard purchased' and sysdate.
 - c. Less 1 from Product_Master.
 - d. Else in purchase_order 'No Keyboard Available' sysdate.
- 23) Accept a number and print factorial of it.

- 24) Accept a number and print it in reverse order.
- 25) Accept a number and print it in reverse order.
- 26) Accept a number and print it is prime or not.
- 27) Accept 10 numbers in loop and print sum of accepted numbers.
- 28) Accept 10 numbers in loop and print sum of accepted even numbers and odd numbers separately.
- 29) Accept a string and print it as follows:
- O
 - Or
 - Ora
 - Orac
 - Oracl
 - Oracle
- 30) Using PL/SQL block generate square and cube of 1 to 10 numbers.
Store the same in a table with columns no, square and cube.

207: Project I

(2 Credits, 4 Lab)

Objectives: Few computing professionals can expect to work in isolation for very much of the time. Software development projects are always implemented by groups of people working together as a team. MCA students therefore need to learn about the mechanics and dynamics of effective team participation as part of their education. Moreover, because the value of working in teams (as well as the difficulties that arise) does not become evident in small-scale projects, students need to engage in team-oriented projects that extend over a reasonably long period of time, possibly a full semester or a significant fraction thereof. To ensure that students have the opportunity to acquire these skills, the MCA programme has four Minor Projects and one full semester project. The course 207 is the first in this series. Students in groups of 3 or 4 (only) are expected to undertake a project preferable in C environment. Every student in the group spends about 50 hours on the project. The guide should simulate the team environment and each student performance their major responsibilities satisfactorily.
