Master of Computer Application (Bridge Course)

Name of Program	**Master of Computer Application (Bridge Course)**
Eligibility Criteria	For students taking admission to MCA course and who have graduated without specialization in computer science/computer application/computer engineering.
Objective of Program	The core objective of the MCA (Bridge Course) program is to bridge the gap between subjects studied at the graduation level and subjects they would be studying in **Master Of Computer Application.**
Program Outcome	The students are expected to be able to understand fundamentals of computer and basic Mathematics
Medium of Instruction	English
Program Structure	

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
101	**Computer Organization and Fundamentals of Operating Systems**	4	0	4	3 Hrs	70	30	100
102	**Data Base Management System**	4	0	4	3 Hrs	70	30	100
103	**Mathematical Foundation Of Computer Science**	4	0	4	3 Hrs	70	30	100
104	**Software Engineering**	4	0	4	3 Hrs	70	30	100
	Total	16	0	16		280	120	400

Note :

(1) The student has to clear the above courses and get 16 credits during the completion of MCA course.

(2) MCA degree shall not be awarded unless the student successfully gets 16 credits for the Bridge Course.

Course: 101: Computer Organization and Fundamentals of Operating Systems

Course Code	101
Course Title	Computer Organization and Fundamentals of Operating Systems
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course is an introduction to students to understand fundamentals of computer. The course also gives students an idea about various components of computer, hardware and its working. The course also explains the concept of various operating systems, their use, architecture and its working.
Course Objective	1. To make students understand computer hardware fundamentals 2. To make students understand various components of computer and their Working 3. To make students understand the importance and use of operating system 4. To explain various types of operating system architecture
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand how computer works and the importance of various components of computers. This course will also help students to appreciate the role of various operating systems. After successful completion, students will be able to select particular configuration of computer and operating system necessary for the application.
Course Content	Unit 1. Memory, Number System & Basic Computer Architecture 1.1 RAM, ROM, PROM, EPROM etc, Virtual Memory, Cache Memory 1.2 Secondary Storage Devices 1.3 Binary, Hexadecimal, Octal Number System 1.4 Integer & Floating Point representations 1.5 Coding Schemes – ASCII, EBCDIC, UNICODE 1.6 Block Diagram of CPU and execution process. 1.7 Introduction to bus architecture 1.8 H/W parts of PC 1.9 Disk Architecture 1.10 I/O devices: Keyboard, Display, Pointing devices, MODEM, Scanners, OMR, OCR, CD-ROM, DVD, printers.
	Unit 2. Operating System Concepts 2.1 Evolution of Operating System & History 2.2 Needs of an Operating System 2.3 Single User & Multi-User Operating System 2.4 Elements of an Operating System Unit 3. CPU Concepts 3.1 BIOS, POST Operation, Types of interrupts, Device Drivers 3.2 IVT 3.3 CISC Vs. RISC Unit 4. Logical Gates 4.1 Concepts of Analog and Digital 4.2 AND & OR gates 4.3 NAND & NOR gates

	4.4 XOR gate 4.5 Flip-Flops 4.6 IC
	Unit 5. Introduction to System S/W Components 5.1 Assemblers 5.2 Compilers 5.3 Linker & Loader [Self Study] Comparison of two multiuser OS
Reference Books	1. Fundamentals of Computer – V. RajaRaman 2. How Computers work - Ron White – Techmedia 3. Introduction to computers:- Peter Norton – TMH 4. Understanding Operating Systems - 4th Ed- Flynn – Thomson Course Technology 5. Inside IBM PC - Peter Norton – PHI 6. Structured Computer Organization - Tanenbaum A.S. – Prentice Hall 7. Digital Computer Electronics – Malvino A.P - TMH 8. Netware for dummies - Dummy Series 9. Computer System Architecture – M.M.Mano - Pearson 10.User Manual of DOS, Windows-Windows-NT, Netware 11.Operating Systems - Stallings – PHI 12.System Programming & Operating System – Dhamdhere- TMH 13.Compilers Principles Techniques & Tools – Aho A – Addison Wesley
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% external assessment

Course: 102: Data Base Management System

Course Code	102
Course Title	Data Base Management System
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course introduces the fundamentals of Database Management System. The course includes the basic of various DBMS's like Access, Mysql and DB2.
Course Objective	1. To make students understand Structured Query Language 2. To get students acquainted with various DBMSs 3. To make students understand and implement the Normalization process.
Pre-requisite	Basic computer fundamentals
Course Outcome	After studying the course, students will be able to understand how to work with DBMS and SQL. This course will also help students to learn normalization and various aspects of Data Base Management System and will also be able to execute SQL queries.
Course Content	Unit 1. Basic Concepts of Database Management System. 1.1 Fundamental concepts of File and databases 1.2 Overview of Physical Storage Media 1.3 Disk Storage 1.4 File Organization 1.5 Organization of records in File 1.6 Purpose of database system 1.7 Introduction to Data models 1.7.1 Conceptual Data model – E- R model 1.7.2 Record based Data models – Hierarchical, Relational, Network 1.8 Features of Database Systems 1.8.1 Data abstraction & Data independence 1.8.2 Type of Database Languages: DDL, DML,TCL 1.8.3 Database users: Database manager, administrator and Users 1.8.4 Overall system structure. 1.8.5. Data – Dictionary 1.9 NOSQL Databases Unit 2. Structure of relational database model and Integrity Constrains 2.1 Relation scheme, Relation, Views 2.2 Notion of Keys 2.3 Pure query language: relational algebra 2.4 Domain Constrains, key Constrains, Referential Integrity Constrains 2.5 Functional Dependencies Unit 3. Relational Commercial Language – SQL 3.1 SQL Constraints 3.1.1 Primary Key 3.1.2 Foreign Key 3.1.3 Not Null / Required Columns 3.1.4 Unique 3.1.5 Check 3.1.6 Default 3.1.7 Auto Increment 3.2 Create, Update, Delete and Read (CRUD) operations

	3.2.1 Create Statement 3.2.2 Alter Statement 3.2.3 Insert Statement 3.2.4 Update Statement 3.2.5 Delete Statement 3.2.6 Select statement and its clauses like Where, Like, In, Between, And, Or, Order By, Group By, Having, Alias, Top, Distinct, Join, Join Types, etc… Unit 4. Commercial RDBMS: Microsoft Access, DB2 & MySQL 4.1 Basic Architecture of DBMS 4.2 Working with databases and tables 4.3 Managing constrains and relationships 4.4 Using SQL queries Unit 5. Relational Database Design 5.1 Pitfalls in relational database design 5.2 Normalization using Functional Dependencies 5.3 Normalization using Multi valued Dependencies 5.4 Normalization using Join Dependencies 5.5 Domain – Key normal form [Self Study] Security features provided by access/DB2
Reference Books	1. Database System Concepts – SILBERSCHATZ, KORTH, SUDARSHAN- McGraw- Hill 2. An introduction to Database Systems- C.J.DATE – Addison Wesley 3. Database System: A practical approach to design implementation and management – THOMAS CONNOLLY, CAROLYN BEGG, Pearson Education 4. Access - The Complete Reference – Virginia Andersen – McGraw – Hill 5. Access Database Design & Programming – Steven Roman - O' Reilly 6. Microsoft Access: Bible – Cary N. Prague
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 104: Mathematical Foundation Of Computer Science

Course Code	104
Course Title	Mathematical Foundation Of Computer Science
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course explores the area of statistics, linear algebra and graph theory. The course has been introduced for learning these concepts to apply them in various algorithms or protocols in the computer relevant subjects in future semesters.
Course Objective	1. To make students learn interpret statistical data 2. To make students learn various statistical concepts like mean, variance, standard deviation, correlation, regression etc. 3. To make students learn linear algebra involving matrix and operations upon Matrices 4. To make students learn about graph theory and its concepts
Pre-requisite	Nil
Course Outcome	After studying the course, students will be able to interpret financial data and perform statistical analysis using mean, variance, standard deviation etc. This course also helps students learn apply linear algebra for matrix and its operations. Students will also be able to use graph theory and its applications in forming matrix and trees.
Course Content	1. Statistics 1.1 Concepts of Random Variable 1.2 Probability & Probability Distribution 1.3 Mean & Variance of Probability Distribution 1.4 Statistical data 1.5 Frequency Distributions 1.6 Mean, Variance & Standard Deviation of Data 1.7 Bivariate data 1.8 Concept of Dependent & Independent Variable 1.9 Correlation 1.10 Linear Regression 2. Linear Algebra 2.1 Matrices 2.2 Elementary matrix operations 2.3 Determinants 2.4 Rank of matrix 2.5 Adjoin & Inverse of a matrix 2.6 Matrix representation of simultaneous linear equation 2.7 Solution of simultaneous linear equations using Cramer's Rule 3. Graph Theory 3.1 Definition & Representation of graphs 3.2 Properties of general Graphs 3.3 Matrix representation of Graphs 3.4 Trees and their properties. 4.[Self Study] Problem solving using Excel/SPSS
Reference Books	1. Introduction to Mathematical Statistics - Hogg R V & Craig A L – Tata Mc-Graw Hill

	2. An Introduction to the Theory of Statistics - Yule U G & Kendall M G - Chailes Griffin & Co. 3. Fundamental of Mathematical Statistics - Kapoor & Gupta - S Chand & Co. 4. Matrix and Linear Algebra - K B Datta - PHI 5. Linear Algebra - A R Rao & P Bhimashankaram - Tata McGraw Hill 6. Graph Theory - Harry F - Addison Wesley Publication
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 105-Software Engineering

Course Code	105
Course Title	Software Engineering
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course is an introduction to students to understand life cycle of Software engineering. The course also gives students an idea about various process models of software engineering
Course Objective	1. To make students understand software engineering 2. To make students understand various components of software engineering 3. To make students understand the importance of software Models
Pre-requisite	Nil
Course Out come	After studying the course, students will be able to understand the life cycle of Software engineering. This course will also help students to appreciate the Software engineering life cycle. After successful completion, students will be able to select particular model of software engineering
Course Content	Unit 1: Introduction 1.1 Role of Software 1.2 What is software? 1.3 Software characteristics 1.4 Software Engineering: definition 1.5 Process models 1.5.1 Waterfall Model 1.5.2 Prototyping 1.5.3 Incremental models 1.5.4 Spiral model 1.6 Agile Processes 1.6.1 Agility Principles 1.6.2 Agile Modelling 1.7 Agile Process Models 1.7.1 Extreme Programming Model 1.7.2 Adaptive Software Development Model 1.7.3 Scrum Model 1.7.4 Feature Driven Development Model 1.7.5 Lean Software Development Model Unit 2: Requirement Engineering and Requirement analysis modelling, Design 2.1 Requirement engineering tasks 2.2 Introduction to requirement Inception, Elicitation, Elaboration, Negotiation, Specification, Validation, Management 2.3 Requirement analysis 2.4 Elements of analysis model and modelling approaches 2.5 Data Flow oriented modelling 2.6Software requirement specification 2.7Translation of analysis model into the design model 2.8 Design process and design quality

2.9 Design concepts
2.10 Design model
2.11 Concept of Pattern based software design
2.12 Introduction: Architecture design
2.13 Mapping data flow into software architecture
2.14 User interface design: golden rules

Unit 3: Software Project management, metrics, Estimation & Scheduling
3.1 People
3.1.1 Stack holders
3.1.2 Team leaders
3.1.3 Software team
3.1.4 Agile teams
3.2 Product
3.2.1 Software scope
3.2.2 Problem decomposition
3.3 Process
3.3.1 Process decomposition
3.4 Project
3.4.1 Issues, approaches
3.5 W5HH principle
3.6 Measures, metrics and indicators
3.7 Process metrics and process improvement
3.8 Project metrics
3.9 Size oriented metrics
3.10 Function oriented metrics
3.11 Metrics for software quality
3.12 Integrating metrics within the software process
3.13 Software project estimation
3.13 Decomposition techniques
3.14 Problem based estimation
3.15 LOC based estimation
3.16 FP based estimation
3.17 Process based estimation
3.18 Empirical estimation models
3.19 Basic concepts
3.20 Basic principles
3.21 Relationship between people and effort
3.22 Effort distribution
3.23 Task network
3.24 Scheduling and tracking
3.25 Earned value analysis

Unit 4: Software Testing & Quality Management
4.1 Testing fundamentals
4.1.1 Principles
4.1.2 Test characteristics
4.2 White box testing
4.2.1 Basis path testing
4.2.2 Control structure testing
4.3 Black box testing
4.3.1 Equivalence partitioning
4.3.2 Boundary value analysis
4.4 Software Testing Strategies
4.4.1 Unit testing
4.4.2 Integration testing
4.4.3 Validation testing
4.4.4 System testing
4.5 Quality concepts

	4.6 Software quality assurance 4.7 SQA activities 4.8 Formal technical reviews 4.9 Statistical software quality assurance 4.10 Software reliability Unit 5: Risk Management 5.1 Strategies 5.2 Software risks 5.3 Risk identification, assessment, components, and drivers 5.4 Risk projection 5.5 Risk mitigation, monitoring and management [Self Study] Test Case Generation and Testing using Testing Tool SRS and case studies should be carried out.
Reference Books	1. Software Engineering – A Practitioner's approach - R.S. Pressman 2. Software engineering principles and practice- Javadekar-McGraw Hill 3. Software engineering -Ian Somerville- Tata McGraw Hill 4. Fundamentals of software engineering,3rd edition-Rajib Mall - Addition Wesley / pearson education PHI 5. An integrated approach to software engineering,3rd Edition-Pankaj Jalote-Narosa 6. Software engineering with java- Stephan R. Schach –TMH
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment