SARVEPALLI RADHAKRISHNAN UNIVERSITY,
BHOPAL (M.P.)

Scheme of Examination

First Semester
Master of Computer Application

Wef:- 2020-21

S.No	Subject Code	Subject Name	Periods per week			Credits	Maximum Marks (Theory Slot)			Maximum Marks (Practical Slot)		Total Marks
			L	T	P		End sem. Exam	Tests (Two)	Assignment /Quiz	End Sem Practical/ Viva	Practical record/ Assignment/Quiz/ Presentation	
1	CA101	Software Engineering.	3	1	-	4	70	20	10		-	100
2	CA102	Programming C & C++	3	1	-	4	70	20	10	-	-	100
3	CA103	Theory of computation	3	1	-	4	70	20	10	-	-	100
4	CA104	Computer Network	3	1	-	4	70	20	10	-	-	100
5	CA105	Data Base Management System	3	1	-	4	70	20	10	-	-	100
6	CA106	Programming Lab of RDBMS	-	-	8	8	-	-	-	120	80	200
7	CA107	Programming Lab In C & C++	-	-	2	2	-	-	-	30	20	50
		Total	15	5	10	30	350	100	50	150	100	750

L: Lecture - T: Tutorial - P: Practical

CA-101 Software Engineering

Objectives of the course:

To expose the students to the following:

1. Provide fundamental knowledge of software engineering
2. To help students to develop skills that will enable them to construct software of high quality – software that is reliable, and that is reasonably easy to understand, modify and maintain
3. Overall system development life cycles (SLDC) provide students to understand the overall system to project according their need.

Total –(60 Hours)

Unit-I (12 Hours) System concepts and Information system environment:

The system concept, characteristics of system, elements of system, The System Development Life Cycle, The Role of System Analyst. Introduction system planning & initial investigation, various information Gathering tools feasibility study conretions & structures tools of system analysis, various methods of Process design, form design methodologies, introduction to information system testing, quality assurance security & destruct computer various (deleting recovery).

Unit-II (12 Hours) Software Process, Product and Project:

The Product: Software, Software Myths, The process: Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Software Process Models, Component – Based Development, Fourth Generation Techniques, Software process and Project Metrics: Software measurement

Unit-III (12 Hours) Software Project Planning and Design:

Software Project Planning: Project planning objectives, Decomposition Techniques, Empirical estimation Models, The Make/Buy Decision., Risk analysis. Software Design: Design Principles, Cohesion & Coupling, Design notation and specification, structure Design methodology.

Unit-IV (12 Hours) Software Quality Assurance and Testing:

Software Quality Assurance: Quality Concepts, The Quality Movement, Software Quality Assurance, Software Reviews, Formal Technical Reviews, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, Mistake Proofing for Software, Introduction to ISO standard. Testing Strategies: A strategic approach of software testing strategic issues, unit testing, integration Testing, validation testing, system testing, the art of debugging. OOA, OOD.

Software Testing Techniques & Stragies : White Box Testing, Basis Path Testing, Control Structure Testing Black Box Testing, Graph Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis, Comparison Testing, Orthogonal Array Testing, Strategic Issues, Unit testing, Integration testing, Validation testing, System Testing, Formal Technical Review.

Unit-V (12 Hours) Advanced Topics:

MIS & DSS: Introduction to MIS, long range planning, development and implementation of an MIS, Applications of MIS in manufacturing sector and in service sector. Decision Support System concepts, types of DSS. Object Oriented Software Engineering: Object Oriented Concepts, Identifying the Elements of an Object Model, Management of Object Oriented Software Projects. CASE tools, Re-engineering

Books

1. R. S. Pressman, "Software Engineering – A practitioner's approach", 6th ed., McGraw Hill Int. Ed., 2002.
2. Pankaj Jalote "Software Engg" Narosa Publications.
3. Ian Sommerville : Software Engineering 6/e (Addison-Wesley)

Course outcomes:-

1. How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment
2. An ability to work in one or more significant application domains
3. Work as an individual and as part of a multidisciplinary team to develop and deliver quality software
4. Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle
5. Demonstrate an ability to use the techniques and tools necessary for engineering practice

CA- 102 Programming with C & C++

Objectives of the course:

To expose the students to the following:

1. The course aims to provide exposure to problem-solving through programming.
2. It aims to train the students to the basic concepts of the C & C++-programming language.
3. This course involves a lab component which is designed to give the student hands-on experience with the concepts.
4. Implement several programs in languages other than the one emphasized in the core curriculum
5. Understand design/implementation issues involved with variable allocation and binding, control flow, types, subroutines, parameter passing
6. Develop an understanding of the compilation process
7. Choose the right data representation formats based on the requirements of the problem.
8. Describe the object-oriented programming approach in connection with C++
9. Apply the concepts of object-oriented programming

Total-(60 Hours)

UNIT-I (9 Hours)

Fundamentals of C Programming: History of C; Structure of a C Program; Data types; Constant & Variable, naming variables; Operators & expressions; Control Constructs – if-else, for, while, do-while; Case switch statement; Arrays; Formatted & unformatted I/O; Type modifiers & storage classes; Ternary Operator; Type conversion & type casting; Priority & associatively of operators.

UNIT-II (9 Hours)

Modular Programming: Functions; Arguments; Return value; Parameter passing – call by value, call by reference; Return statement; Scope, visibility and life-time rules for various types of variable, static variable; Calling a function; Recursion – basics, comparison with iteration, types of recursion-direct, indirect, tree and tail recursion, when to avoid recursion, examples.

UNIT-III (9 Hours)

Advanced Programming Techniques: Special constructs – Break, continue, exit(), go to & labels; Pointers - & and * operators, pointer expression, pointer arithmetic, dynamic memory management functions like malloc(), calloc(), free(); String; Pointer v/s array; Pointer to pointer; Array of pointer & its limitation; Function returning pointers; Pointer to function, Function as parameter; Structure – basic, declaration, membership operator, pointer to structure, referential operator, self referential structures, structure within structure, array in structure, array of structures; Union – basic, declaration; Enumerated data type; Type def; command line arguments.

UNIT-IV (9 Hours)

Overview of C++: Object oriented programming, Concepts, Advantages, Usage. C++ Environment: Program development environment, the language and the C++ language standards. Introduction to various C++ compilers, C++ standard libraries, Prototype of main () function, Data types. C++ as a superset of C, New style comments, main function in C++, meaning of empty argument list, function prototyping, default arguments and argument matching. User defined data types: enumerated types, use of tag names, anonymous unions, scope of tag names Classes & Objects : Classes, Structure & Classes, Union & Classes, Inline Function, Scope Resolution operator, Static Class Members: Static Data Member, Static Member Function, Passing Objects to Function, Returning Objects, Object Assignment. Friend Function, Friend Classes

Unit-V (9 Hours)
Array, Pointers References & The Dynamic Allocation Operators: Array of Objects, Pointers to Object, Type Checking C++ Pointers, The This Pointer, Pointer to Derived Types, Pointer to Class Members, References: Reference Parameter, call by reference and return by reference Passing References to Objects, Returning Reference, Independent Reference, C++'S Dynamic Allocation Operators, Initializing Allocated Memory, Allocating Array, Allocating Objects. Constructor & Destructor : Introduction, Constructor, access specifiers for constructors, and instantiation, Parameterized Constructor, Multiple Constructor in A Class, Constructor with Default Argument, Copy Constructor, Destructor

Unit-VI (9 Hours)
Overloading as polymorphism: Function & Operator Overloading : Function Overloading, Overloading Constructor Function Finding the Address of an Overloaded Function, Operator Overloading: Creating A Member Operator Function, Creating Prefix & Postfix Forms of the Increment & Decrement Operation, Overloading The Shorthand Operation (I.E. +=,-= Etc), Operator Overloading Restrictions, Operator Overloading Using Friend Function, Overloading New & Delete, Overloading Some Special Operators, Overloading [ ], ( ), -, Comma Operator, Overloading << And . Namespaces: global namespace and namespace std, nested namespaces.

Unit-VII (6 Hours)
Inheritance : Base Class Access Control, C, Protected Base Class Inheritance, Inheriting Multiple Base Classes, Constructors, Destructors & Inheritance, When Constructor & Destructor Function are Executed, Passing Parameters to Base Class Constructors, Granting Access, Virtual Base Classes. Virtual Functions & Polymorphism : Virtual Function, Pure Virtual Functions, Early Vs. Late Binding

Books
1. Kanetkar Y. "Let us C", BPB.
2. Kanetkar Y.: "Pointers in C" , BPB
3. Lafore R. "Object Oriented Programming in C++", Galgotia Pub.
4. Schildt "C++ the complete reference 4ed, 2003.
5. Balagurusawmy "Object Oriented Programming with C++".

Outcomes

After the course the students are expected to be able to (this is what the exams will test)

1. Identify situations where computational methods and computers would be useful
2. Given a computational problem, identify and abstractthe programming task involved.
3. Use the comparisons and limitations of the various programming constructs and choose the right one for the task in hand.
4. Write the program on a computer, edit, compile, debug, correct, recompile and run it.
5. Identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task.
6. Understand the difference between the top-down and bottom-up approach
7. Illustrate the process of data file manipulations using C++
8. Apply virtual and pure virtual function & complex programming situations

CA-103 Theory of Computation

Objectives of the course:

To expose the students to the following:

1. To learn fundamentals of Regular and Context Free Grammars and Languages
2. To understand the relation between Regular Language and Finite Automata and machines.
3. To learn how to design Automata's and machines as Acceptors, Verifiers and Translators.
4. To understand the relation between Contexts free Languages, PDA and TM.
5. To learn how to design PDA as acceptor and TM as Calculators.

Total-(60 Hours)

Unit-I (12 Hours)

Review of Mathematical Preliminaries: Set, Relations and functions, Graphs and trees, string, alphabets And languages. Principle of induction, predicates and propositional calculus.
Theory of Automation: Definition, description, DFA, NFA, Transition systems, 2DFA, equivalence of DFA & NDFA, Regular expressions, regular grammar, FSM with output (mealy and moore models), Minimization of finite automata.

Unit-II (12 Hours)

Formal Languages: Definition & description, Pharse structured grammars & their classification, Chomskey classification of languages, closure properties of families of language, regular grammar, Regular set & their closure properties, finite automata, equivalence of FA and regular expression, Equivalence of two way finite automata, equivalence of regular expressions.

Unit-III (12 Hours)

Context-Free grammar & PDA: Properties unrestricted grammar & their equivalence, derivation tree Simplifying CFG, unambiguifying CFG, ε-productions, normal form for CFG, Pushdown automata, 2 Way PDA, relation of PDA with CFG, Determinism & Non determinism in PDA & related theorems, Parsing and pushdown automata.

Unit-IV (12 Hours)

Turing Machine: Model, design, representation of TM, language accepted by TM, universal turing Machine, determine & non-determinism in TM, TM as acceptor/generator/algorithms, multidimensional, Multitracks, multitape, Two way infinite tape, multihead, Halting problems of TM.

Unit-V (12 Hours)

Computability: Concepts, Introduction to complexity theory, Introduction to undecidaibility, recursively Enumerable sets, primitive recursive functions, recursive set, partial recursive sets, concepts of linear Bounded Automata, context sensitive grammars & their equivalence

1. Understand, design, construct, analyze and interpret Regular languages, Expression and Grammars.
2. Design different types of Finite Automata and Machines as Acceptor, Verifier and Translator.
3. Understand, design, analyze and interpret Context Free languages, Expression and Grammars.
4. Design different types of Push down Automata as Simple Parser.
5. Design different types of Turing Machines as Acceptor, Verifier, Translator and Basic computing machine.
6. Compare, understand and analyze different languages, grammars, Automata and Machines.

Books

1. Hopcroft & Ullman “Introduction to Automata theory, languages & Computation” , Narosha Publishing house.
2. Peter Linz, “An Introduction to formal language and automata”, Third edition, Narosa publication.
3. Marvin L. Minskay “Computation: Finite & Infinite Machines”, PHI.
4. Mishra & Chander Shekhar “Theory of Computer Science (Automate, Language & Computations), PHI.

CA-104 Computer Networks

Course Objectives:

1. Study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
2. Acquire knowledge of Application layer and Presentation layer paradigms and protocols.
3. Study Session layer design issues, Transport layer services, and protocols.
4. Gain core knowledge of Network layer routing protocols and IP addressing.
5. Study data link layer concepts, design issues, and protocols.
6. Read the fundamentals and basics of Physical layer, and will apply them in real time applications.

Total-60 Hours

UNIT-I (12 Hours)

Introduction: Computer Network, Layered Network Architecture-Review of ISO-OSI Model., Transmission Fundamentals- Communication Media- Conductive Metal (Wired Cable), Optical Fiber links, Wireless Communication-Radio links, Satellite Links, Communication Services & Devices, Telephone System., Integrated Service Digital Network (ISDN)., Cellular Phone., ATM, Modulation & Demodulation-, Digital to Analog Conversion-Frequency Modulation (FM), Amplitude, Modulation (AM), Phase Modulation (PM).,Analog to Digital Conversion-Pulse Amplitude Modulation(PAM), Pulse Code Modulation (PCM),Differential Pulse Code Modulation, (DPCM)., Modem & Modem Types., Multiplexing-, Frequency Division Multiplexing (FDM)., Time Division Multiplexing (TDM), Statistical Time Division Multiplexing(STDM)., Contention Protocol-, Stop-Go-Access Protocol, Aloha Protocol-Pure aloha & Slotted aloha, Carrier sense multiple access with collision detection (CSMA/CD).

UNIT-II (12 Hours)

Data Security and Integrity: Parity Checking Code, Cyclic redundancy checks (CRC), Hemming Code, Protocol Concepts –, Basic flow control, Sliding window protocol-Go-Back-N protocol and selective repeat protocol, Protocol correctness- Finite state machine.

UNIT-III (12 Hours)

Local Area Network: Ethernet: 802.3 IEEE standards, Token Ring: 802.5 IEEE standard, Token Bus: 802.4 IEEE standard, FDDI Protocol, DQDB Protocol, Inter Networking, Layer 1 connections-Repeater, Hubs, Layer 2 connections- Bridges, Switches, Layer 3 connections- Routers, Gateways. Introduction to Router, Configuring a Router, Interior & Exterior Routing, RIP, Distance Vector Routing, OSPF, BGP, Uni-cast, Multicast and Broadcast. Multicast routing protocols: DVMRP, MOSPF, CBT, PIM, MBONE, EIGRP, CIDR, Multicast Trees, Comparative study of IPv6 and IPv4

UNIT-IV (12 Hours)

Wide Area Network: Introduction, Network routing, Routing Tables, Types of routing, Dijkstra's Algorithm, Bellman-Ford Algorithm, Link state routing, Open shortest path first, Flooding, Broadcasting, Multicasting, Congestion & Dead Lock, Internet Protocols, Overview of TCP/IP, Transport protocols, Elements of Transport Protocol, Transmission control protocol (TCP), User data-gram protocol (UDP).

UNIT-V (12 Hours)

Network Security: Virtual Terminal Protocol, Overview of DNS, SNMP, email, WWW, Multimedia.

Course Outcomes:

Students will able to:

1. Describe the functions of each layer in OSI and TCP/IP model.
2. Explain the functions of Application layer and Presentation layer paradigms and Protocols.
3. Describe the Session layer design issues and Transport layer services.
4. Classify the routing protocols and analyze how to assign the IP addresses for the given network.
5. Describe the functions of data link layer and explain the protocols. 6. Explain the types of transmission media with real time applications.

BOOKS:

1. A.S.Tanenbaum, “Computer Network”, 4th addition, PHI
2. Forouzan “Data Communication and Networking 3ed”, TMH
4. D.E.Comer, “Internetworking with TCP/IP”, Volume Ist & IInd, PHI
5. Willium Stalling, “Data & Computer communications”, Maxwell Macmillan International Ed

CA-105 Data Base Management System

Objectives of the course

1. To understand the different issues involved in the design and implementation of a database system.
2. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
3. To understand and use data manipulation language to query, update, and manage a database To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
4. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Total- (60 Hours)

Unit-I (12 Hours)

Introduction: Advantage of DBMS approach, various view of data, data independence, schema and subschema, Primary concepts of data models, Database languages, transaction management, Database Administrator and users, data dictionary, overall system architecture.ER model: basic concepts, design issues, mapping constraint, keys, ER diagram, weak and strong entity Sets, specialization and generalization, aggregation, inheritance, design of ER schema, reduction of ER Schema to tables.

Unit-II (12 Hours)

Domains, Relations and Keys: domains, relations, kind of relations, relational database, various types of Keys, candidate, primary, alternate and foreign keys.
Relational Algebra & SQL: The structure, relational algebra with extended operations, modifications of Database, idea of relational calculus, basic structure of SQL, set operations, aggregate functions, null Values, nested sub queries, derived relations, views, modification of Database, join relations, DDL in SQL.

Unit-III (12 Hours)

Functional Dependencies and Normalization: basic definitions, trivial and non trivial dependencies, Closure set of dependencies and of attributes, irreducible set of dependencies, introduction to normalization, non loss decomposition, FD diagram, first, second, third Normal forms, dependency Preservation, BCNF, multivalued dependencies and fourth normal form, Join dependency and fifth normal Form.

Unit-IV (12 Hours)

Database Integrity: general idea. Integrity rules, domain rules, attribute rules, relation rules, Database Rules, assertions, triggers, integrity and SQL.
Transaction, concurrency and Recovery: basic concepts, ACID properties, Transaction states, Implementation of atomicity and durability, concurrent executions, basic idea of serializability, basic idea Of concurrency control, basic idea of deadlock, failure classification, storage structure types, stable Storage implementation, data access, recovery and atomicity- log based recovery, deferred Database Modification, immediate Database modification, checkpoints.
Distributed Database: basic idea, distributed data storage, data replication, data fragmentation horizontal, Vertical and mixed fragmentation

Unit-V (12 Hours)

Emerging Fields in DBMS: object oriented Databases-basic idea and the model, object structure, object Class, inheritance, multiple inheritance, object identity, data warehousing- terminology, definitions, Characteristics, data mining and its overview, Database on www, multimedia Databases-difference with Conventional DBMS, issues, similarity based retrieval, continuous media data, multimedia data formats, Video servers.Storage structure and file organizations: overview of physical storage media, magnetic disks performance And optimization, basic idea of RAID, file organization, organization of records in files, basic Concepts of indexing, ordered indices, basic idea of B-tree and B+-tree organization Network and hierarchical models: basic idea, data structure diagrams, DBTG model, implementations, tree structure diagram, implementation techniques, comparison of the three models
Web databases: Accessing databases through web

Course Outcomes

1. For a given query write relational algebra expressions for that query and optimize the developed expressions
2. For a given specification of the requirement design the databases using ER method and normalization.
3. For a given specification construct the SQL queries for Open source and Commercial DBMS -MYSQL, ORACLE, and DB2.
4. For a given query optimize its execution using Query optimization algorithms
5. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.
6. Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Books

1. A Silberschatz, H.F Korth, Sudersan “Database System Concepts” –, MGH Publication.
2. C.J Date “An introduction to Database Systems” –6th ed.
3. Elmasri & Navathe “Fundamentals of Database systems” – III ed.