School of Information Technology
Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal
New Scheme of Examination as per AICTE Flexible Curricula
Syllabus II Semester
B.Tech Computer Science and Engineering (Artificial Intelligence and Machine Learning)

AL-201 (DISCRETE STRUCTURES)

Objective-This course introduces the applications of discrete mathematics in the field of computer science. It covers sets, logic, proving techniques, combinatorics, functions, relations, Graph theory and algebraic structures. These basic concepts of sets, logic functions and graph theory are applied to Boolean Algebra and logic networks while the advanced concepts of functions and algebraic structures are applied to finite state machines and coding theory.

Unit-I

Set Theory, Relation, Function, Theorem Proving Techniques : Set Theory: Definition of sets, countable and uncountable sets, Venn Diagrams, proofs of some general identities on sets Relation: Definition, types of relation, composition of relations, Pictorial representation of relation, Equivalence relation, Partial ordering relation, Job-Scheduling problem Function: Definition, type of functions, one to one, into and onto function, inverse function, composition of functions, recursively defined functions, pigeonhole principle. Theorem proving Techniques: Mathematical induction, Proof by contradiction.

Unit-II

Algebraic Structures: Definition, Properties, types: Semi Groups, Monoid, Groups, Abelian group, properties of groups, Subgroup, cyclic groups, Cosets, factor group, Permutation groups, Normal subgroup, Homomorphism and isomorphism of Groups, example and standard results, Rings and Fields: definition and standard results.

Unit-III

Propositional Logic: Proposition, First order logic, Basic logical operation, truth tables, tautologies, Contradictions, Algebra of Proposition, logical implications, logical equivalence, predicates, Normal Forms, Universal and existential quantifiers. Introduction to finite state machine Finite state machines as models of physical system equivalence machines, Finite state machines as language recognizers.

Unit-IV

Graph Theory: Introduction and basic terminology of graphs, Planer graphs, Multigraphs and weighted graphs, Isomorphic graphs, Paths, Cycles and connectivity, Shortest path in weighted graph, Introduction to Eulerian paths and circuits, Hamiltonian paths and circuits, Graph coloring, chromatic number, Isomorphism and Homomorphism of graphs.

Unit V

Posets, Hasse Diagram and Lattices: Introduction, ordered set, Hasse diagram of partially, ordered set, isomorphic ordered set, well ordered set, properties of Lattices, bounded and complemented lattices. Combinatorics: Introduction, Permutation and combination, Binomial Theorem, Multimonial Coefficients Recurrence Relation and Generating Function: Introduction to Recurrence Relation and Recursive algorithms , Linear recurrence relations with constant coefficients, Homogeneous solutions, Particular solutions, Total solutions , Generating functions, Solution by method of generating functions.

References:

1. C.L.Liu, “Elements of Discrete Mathematics” Tata Mc Graw-Hill Edition.
2. Trembley, J.P & Manohar, “Discrete Mathematical Structure with Application CS”, McGraw Hill.
3. Kenneth H. Rosen, “Discrete Mathematics and its applications”, McGraw Hill.
4. Bisht, “Discrete Mathematics”,Oxford University Press

5. Biswal,”Discrete Mathematics & Graph Theory”, PHI

Course Outcomes:

After completion of the course students would be able to:
CO1: Prove mathematical theorems using mathematical induction.
CO2: Understand sets and perform operations and algebra on sets.
CO3: Demonstrate an understanding of relations and functions and be able to determine their properties.
CO4: Understand groups, rings and fields.
CO5: Analyze logical propositions via truth tables.
CO6: Define graphs, digraphs and trees, and identify their main properties.
CO7: Evaluate combinations and permutations on sets.

SCHOOL OF INFORMATION TECHNOLOGY, RGPV, BHOPAL

AL-202 (STATISTICAL METHODS)

Unit I- Linear Statistical Models: Simple linear regression & correlation, multiple regression & multiple correlation, Analysis of variance (one way, two way with as well as without interaction)

Unit II- Estimation and Sufficient Statistic: Estimation: Point estimation, criteria for good estimates (un-biasedness, consistency), Methods of estimation including maximum likelihood estimation. Sufficient Statistic: Concept & examples, complete sufficiency, their application in estimation

Test of hypothesis: Concept & formulation, Type I and Type II errors, Neyman Pearson lemma, Procedures of testing,

Unit III- Non-parametric Inference: Comparison with parametric inference, Use of order statistics. Sign test, Wilcoxon signed rank test, Mann-Whitney test, Run test, Kolmogorov-Smirnov test. Spearman‘s and Kendall‘s test. Tolerance region

Unit IV- Basics of Time Series Analysis & Forecasting: Stationary, ARIMA Models: Identification, Estimation and Forecasting.

Unit V- R statistical programming language: Introduction to R, Functions, Control flow and Loops, Working with Vectors and Matrices, Reading in Data, Writing Data, Working with Data, Manipulating Data, Simulation, Linear model, Data Frame, Graphics in R

Reference Books-

1. I.R. Miller, J.E. Freund and R. Johnson, "Probability and Statistics for Engineers" 9th Edition, Pearson.
2. A. Goon, M. Gupta and B. Dasgupta, ―Fundamentals of Statistics, vol. I & II, World Press.
3. Chris Chatfield, "The Analysis of Time Series: An Introduction, 6th edition, Chapman and Hall/CRC.
4. D.C. Montgomery &E.Peck, "Introduction to Linear Regression Analysis", 5th edition, Wiley.
5. A.M. Mood, F.A. Graybill& D.C. Boes, ―Introduction to the Theory of Statistics, 3rd edition, McGraw Hill.
6. N. Draper & H. Smith, "Applied Regression Analysis‖, 3rd edition, Wiley.
7. Garrett Grolemund, "Hands-on Programming with R", 1st edition, O′Reilly. 8. Jared P. Lander, "R for Everyone: Advanced Analytics and Graphics", 2nd edition, Addison-Wesley Professional.

Course Outcomes-

On successful completion of the course, the students will be able to:
CO1: Understand various linear statistical models and acquire knowledge in hypothesis testing.
CO2: Apply methods of estimation in statistical analysis.
CO3: Understand Non-Parametric tests and its applications.
CO4: Design and forecast models using Tim series data.
CO5: Understand and apply R language in data visualization.

Suggested List of Experiments-

1. Use of various functions in R
2. R programming

SCHOOL OF INFORMATION TECHNOLOGY, RGPV, BHOPAL

AL-203 (DIGITAL LOGIC DESIGN)

Unit-I

Introduction to Digital Electronics, Needs and Significance, Different Number System: Binary Numbers, Octal and Hexadecimal Numbers, Conversions, Complement's, Signed Binary Numbers, Binary Arithmetic's, Binary Codes: BCD, ASCII Codes.

Unit-II

Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Boolean Relations, Digital Logic Gates, De Morgan's Theorem, Karnaugh Maps and simplifications.

Unit-III

Combinational Circuits, Half Adder, Full Adder, Binary Adder-Subtractor, Binary Multiplier, Comparator, Decoders, Encoders, Multiplexers, Programmable Devices: Programmable Logic Array (PLA), Programmable Array Logic (PAL).

Unit-IV

Sequential Circuits, Latches, Flip-Flops: RS Latches, Level Clocking, D Latches, Edge-triggered D Flip-flop, Edge-triggered JK Flip-flop, JK Master-slave Flip-flop, Registers, Shift Registers, Counters, Ripple Counters, Synchronous Counters.

Unit-V

Introduction to Memory, Memory Decoding, Error Detection and Correction, Sequential (or simple) programmable logic device (SPLD), Complex programmable logic device (CPLD), Field-programmable gate array (FPGA), Digital Logic Design: RTL and DTL Circuits, TTL, ECL, MOS, CMOS, Application Specific Integrated Circuits.

Reference Books-

1. M. Morris Mono, "Digital logic design", Pearson Education Pvt. Ltd.
2. A Anand Kumar, "Fundamentals of digital circuits", PHI Learning Pvt Ltd.
3. A K Maini, "Digital Electronics Principles and Integrated Circuits, Wiley India Pvt Ltd.
4. R P Jain, "Modern Digital Electronics", Tata McGraw-Hill publishing company Ltd.
5. D P Kothari and J S Dhillon, "Digital Circuits and Design", Pearson Education Pvt. Ltd.

Course Outcomes-

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

1. Perform number base conversions
2. Use Boolean logic to create digital circuits and reduce the Boolean functions to mitigate hardware complexity issues
3. Learn design of combinational circuits
4. Learn sequential circuits and use them in digital systems such as computers and communication systems
5. Compare and differentiate various memories used in Computers

Suggested List of Experiments-

1. Study and verify the operation of AND, OR, NOT, NOR and NAND logic gates.
2. Design all basic logic gates using NOR universal gate.

3. Design all basic logic gates using NAND universal gate.
4. Verification of Demorgan’s theorem.
5. Construction and verification of half adder and full adder circuits.
6. Construction and verification of half subtractor and full subtractor circuits.
7. Design of Binary to Grey & Grey to Binary code Converters .
8. Design of BCD to excess-3 code converter.

SCHOOL OF INFORMATION TECHNOLOGY, RGPV, BHOPAL

AL-204 (OBJECT ORIENTED PROGRAMMING)

Unit I

Introduction: Object oriented programming, Introduction, Application, characteristics, difference between object oriented and procedure programming, Comparison of C and C++, Cout, Cin, Data Type, Type Conversion, Control Statement, Loops, Arrays and string arrays fundamentals, Function, Returning values from functions, Reference arguments, Overloaded function, Inline function, Default arguments, Returning by reference.

Unit II

Object and Classes: Implementation of class and object in C++, access modifiers, object as data type, constructor, destructor, Object as function arguments, default copy constructor, parameterized constructor, returning object from function, Structures and classes, Classes objects and memory, static class data, Arrays of object, Arrays as class Member Data, The standard C++ String class, Run time and Compile time polymorphism.

Unit III

Operator overloading and Inheritance: Overloading unary operators, Overloading binary operators, data conversion, pitfalls of operators overloading, Concept of inheritance, Derived class and base class, access modifiers, types of inheritance, Derived class constructors, member function, public and private inheritance.

Unit IV

Pointer and Virtual Function: Addresses and pointers, the address-of operator & pointer and arrays, Pointer and Function pointer, Memory management: New and Delete, pointers to objects, debugging pointers, Virtual Function, friend function, Static function, friend class, Assignment and copy initialization, this pointer, dynamic type information.

Unit V

Streams and Files: Streams classes, Stream Errors, Disk File I/O with streams, file pointers, error handling in file I/O with member function, overloading the extraction and insertion operators, memory as a stream object, command line arguments, printer output, Function templates, Class templates Exceptions, Containers, exception handling.

Reference Books-

1. E. Balaguruswami, ”Object Oriented Programming in C++”, TMH.
2. Robert Lafore, “Object Oriented Programming in C++”, Pearson.
3. M.T. Somashekare, D.S. Guru, “ Object-Oriented Programming with C++”, PHI.
4. Herbert Shildt, “The Complete Reference C++”, Tata McGraw Hill publication.

Course Outcomes-

After the completion of this course, the students will be able to:
CO1: Understand the key features of Object Oriented Programming and Methodology
CO2: Recognize attributes and methods for given objects.
CO3: Implement programs of inheritance and operator overloading
CO4: Understand and use concept of pointers, memory management and virtual functions in programs
CO5: Perform file handling in programs

Suggested List of Experiments-

1. Write a program to find out the largest number using function.
2. Write a program to find the area of circle, rectangle and triangle using function overloading.
3. Write a program to implement complex numbers using operator overloading and type conversion.
4. Write a program using class and object to print bio-data of the students.
5. Write a program which defines a class with constructor and destructor which will count number of object created and destroyed.
6. Write a program to implement single and multiple inheritances taking student as the sample

base class.

7. Write a program to add two private data members using friend function.
8. Write a program using dynamic memory allocation to perform 2x2 matrix addition and subtraction.
9. Write a program to create a stack using virtual function.
10. Write a program that store five student records in a file.
11. Write a program to get IP address of the system.
12. Write a program to shutdown the system on windows operating system.

AL-205 (INTRODUCTION TO ARTIFICIAL INTELLIGENCE)

Unit I Artificial Intelligence: Introduction, History of AI, AI Problem, Approaches, Goals, Purpose, Scope, Terminology, and Application Areas, Industrialization and its Impact, Cyber-Physical System, Evolution of Industry, Data Availability, Relation between Artificial Intelligence, Machine Learning, Deep Learning and other Related Fields.

Unit II Knowledge Representation, Problems in representing knowledge, knowledge representation using propositional and predicate logic, comparison of propositional and predicate logic, Resolution, refutation, deduction, theorem proving, inferencing, monotonic and non-monotonic reasoning.

Unit III Probabilistic reasoning, Baye's theorem, semantic networks, scripts, schemas, frames, conceptual dependency, fuzzy logic, forward and backward reasoning.

Unit IV Game playing techniques like minimax procedure, alpha-beta cut-offs etc, planning, Study of the block world problem in robotics, Introduction to understanding, natural language processing.

Unit V Introduction to learning, various techniques used in learning, Introduction to neural networks, applications of neural networks, common sense, reasoning, some example of expert systems.

Reference Books-

1. Artificial Intelligence A Modern Approach by Stuart J. Russell and Peter Norvig, Prentice Hall.
2. Fundamentals of Artificial Intelligence by K. R. Chowdhary, Springer.
3. E. Alpaydin, Introduction to Machine Learning (3rd ed.), PHI, 2015. ISBN 978-8120350786.

Course Outcomes-

After completion of the course students would be able to:

CO1. Define basic concepts of Artificial Intelligence.
CO2. Compare different data types
CO3. Understand basic concepts of Machine Learning.
CO4. Explain various supervised and unsupervised learning approaches
CO5. Compare various machine learning models

SCHOOL OF INFORMATION TECHNOLOGY, RGPV, BHOPAL

AL-206 (Computer Programming II)
(C++)

UNIT I

Principles of Object Oriented Programming (OOP), Software Evaluation, A Look at ProcedureOriented Programming, OOP Paradigm, Basic Concepts of OOP, Benefits of OPP, Application of OOP.

UNIT II

Introduction to C++

What is C++, A simple C++ Program, More C++ statements, Structure of C++ Program.

Tokens, Expression and controls Structures, Tokens , Keywords, Identifiers and Constants, C++ data types, Variables: Declaration, Dynamic initialization of variables, Reference variables, Operators in C++ : Scope resolution operator, Member, deferencing Operators, Memory Management Operators, Manipulators, Type cast operators, Expressions and Control Structures.

Functions

The main() function, Function Prototyping, Call by reference, Return by reference, Inline function, Function Overloading.

UNIT III

Classes and Objects

Introduction, Specifying a Class, Defining member Functions, C++ Program with Class, Nesting of Member functions, Private member functions, Memory Allocation for Objects, Static Data members, Static Member Functions, Arrays within a Class, Arrays of Objects, Objects as Function Arguments, Friendly Functions, Returning Objects.

Pointers

Pointers : Declaration and initializing, Manipulation of pointers, pointers Expression and Pointer Arithmetic, Pointer with Arrays, Arrays of Pointers, Pointers to objects, this pointers, Arrays of Pointers to Objects

Constructors and Destructors

Constructors, Parameterized Constructors, Multiple Constructors in a class, Copy constructor, Destructors.

Operator overloading

Defining Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Type Conversions.

UNIT IV

Inheritance and Polymorphisms

Introduction, Defining Derived Classes, Single inheritance, Multiple inheritance, Hierarchical inheritance, Multilevel inheritance, Hybrid inheritance, Virtual Base Classes, Polymorphism, static and dynamic binding, Constructor in Derived Classes, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions.

UNIT V

I/O Operations and Files

C++ Stream Classes, Unformatted I/O Operations, Formatted I/O operations, Classes for File Streams, Opening and Closing a File : open() and close() functions, Manipulators of File Pointers : seekg(), seekp(), tellg(), tellp() functions, Sequential Input and output Operations : put (), get(), write(), read() functions, Error handling File Operations : eof(), fail(), bad(), good() .

Reference Books:

1. E. Balagurusamy - Object Oriented Programming with C++ - TMH.
2. Robert Lafore - Object Oriented Programming in Microsoft C++ - Galgotia.

Suggested List of Experiments:

1. Write a C++ Program to display Names, Roll No., and grades of 3 students who have appeared in the examination. Declare the class of name, Roll No. and grade. Create an array of class objects. Read and display the contents of the array.
2. Write a C++ program to declare Struct. Initialize and display contents of member variables.
3. Write a C++ program to declare a class. Declare pointer to class. Initialize and display the contents of the class member.
4. Given that an EMPLOYEE class contains following members: data members: Employee number, Employee name, Basic, DA, IT, Net Salary and print data members.
5. Write a C++ program to read the data of N employee and compute Net salary of each employee (DA=52% of Basic and Income Tax (IT) =30% of the gross salary).
6. Write a C++ to illustrate the concepts of console I/O operations.
7. Write a C++ program to use scope resolution operator. Display the various values of the same variables declared at different scope levels.
8. Write a C++ program to allocate memory using new operator.
9. Write a C++ program to create multilevel inheritance. (Hint: Classes A1, A2, A3)
10. Write a C++ program to create an array of pointers. Invoke functions using array objects.
11. Write a C++ program to use pointer for both base and derived classes and call the member function. Use Virtual keyword
12. Write a Program to overload operators like *, <<, >> using friend function. The following overloaded operators should work for a class vector.
13. Write a program in C++ to highlight the difference between overloaded assignment operator and copy constructor.
14. Write a program to implement the exception handling with re throwing in exception.