Kurukshetra University, Kurukshetra

('A++' Grade, NAAC Accredited)

Bachelor of Technology (Information Technology)

Credit-Based Scheme of Studies/Examination

Semester III & IV (w.e.f. session 2025-2026)

Bachelor of Technology (Information Technology)										
Credit-Based Scheme of Studies/Examination										
Semester III(w.e.f. session 2025-2026)										
S. No.	**Course Code**	**Subject**	**L:T: P**	**Hours/Week**	**Credits**	**Examination Schedule (Marks)**				**Duration of Exam (Hrs)**
						End Semester Exam	**Internal Assessment**	**Practical Exam**	**Total**	
1	B24-IT-201	IT Workshop (Python)	3:0:0	3	3	70	30	0	100	3
2	B24-ESC-IT-215	Digital Electronics and logic design	3:0:0	3	3	70	30	0	100	3
3	B24-IT-203	Data Structure	4:0:0	4	4	70	30	0	100	3
4	B24-IT-205	Object Oriented Programming using C++	4:0:0	4	4	70	30	0	100	3
5	B24-BSC-IT-217	Probability and Statistics	3:0:0	3	3	70	30	0	100	3
6	B24-IT-207	Computer Organization & Architecture	3:0:0	3	3	70	30	0	100	3
7	B24-IT-209	IT Workshop (Python) Lab	0:0:2	2	1	0	40	60	100	3
8	B24-ESC-IT-219	Digital Electronics and logic design Lab	0:0:2	2	1	0	40	60	100	3
9	B24-IT-211	Object Oriented Programming Lab	0:0:3	3	1.5	0	40	60	100	3
10	B24-IT-213	Data Structure Lab	0:0:3	3	1.5	0	40	60	100	3
Total				**30**	**25**	**420**	**340**	**240**	**1000**	

Bachelor of Technology (Information Technology)										
Credit-Based Scheme of Studies/Examination										
Semester IV(w.e.f. session 2025-2026)										
S. No.	Course Code	Subject	L:T:P	Hours/ Week	Credits	Examination Schedule (Marks)				Duration of Exam (Hrs)
						End Semester Exam	Internal Assessment	Practical Exam	Total	
1	B24-IT-202	Principles of Programming Languages	4:0:0	4	4	70	30	0	100	3
2	B24-IT-204	Operating System	3:0:0	3	3	70	30	0	100	3
3	B24-IT-206	Advance Programming (Java)	4:0:0	4	4	70	30	0	100	3
4	B24-IT-208	Database Management Systems	4:0:0	4	4	70	30	0	100	3
5	B24-HSM-IT-216	Organizational Behavior	3:0:0	3	3	70	30	0	100	3
6	B24-IT-210	Advance Programming Lab (Java)	0:0:3	3	1.5	0	40	60	100	3
7	B24-IT-212	Operating System Lab	0:0:2	2	1	0	40	60	100	3
8	B24-IT-214	Database Management Systems Lab	0:0:3	3	1.5	0	40	60	100	3
9	B24-MC-IT-218	Environmental Sciences	3:0:0	3	1	70	30	0	100	3
Total				**29**	**23**	**420**	**300**	**180**	**900**	

Note: All students have to undertake the industrial training for 6 to 8 weeks after 4th semester which will be evaluated in 5th semester.

B24-IT-201	IT Workshop (Python)						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	To familiarize the students with the basics of Python Programming						
	Course Outcomes						
CO1	To Study Fundamental concept of Python.						
CO2	To Study and implement expression and Strings methods						
CO3	To Study and implement tulles, list and dictionary operations.						
CO4	To Study and implement exception handling and file operation.						

Unit-1

Familiarization with the basics of Python programming: Introduction to Python, Features of Python, Execution modes: interactive mode and script mode, Python character set, use of indentation, Python tokens(keyword, identifier, literal, operator, punctuator), variables, use of comments, Knowledge of data types: Number(Integer, Floating point, Complex). Errors: syntax errors, logical errors, and run-time errors

Unit-2

Expressions: Statement, Type conversion, and input/output: Precedence of Operators, Arithmetic operators, relational operators, logical operators, assignment operators, augmented assignment operators, identity operators (is, is not),Expression, evaluation of an expression, type-conversion, Flow of Control,Conditional statements, Iterative Statements Strings: Introduction, string operations (concatenation, repetition, membership and slicing), traversing a string using loops, built-in functions/methods–len(), capitalize(), title(), lower(), upper(), count(), find(), index(), endswith(), startswith(), isalnum(), isalpha(), isdigit(), islower(), isupper(), isspace(),lstrip(), rstrip(), strip(), replace(), join(), partition(), split()

Unit-3

Array: Access the Elements of an Array, Length of an Array, Adding Array Elements, Removing Array Elements, Adds and remove the element at the specified position. Lists, Tuples, Dictionary: introduction, indexing, list operations, traversing a list using loops, built-in functions/methods–len(), list(), append(), extend(), insert(), count(), index(), remove(), pop(), reverse(), sort(), sorted(), min(), max(), sum(). Introduction to Python modules: Importing module using ‘import ’ and using from statement, importing math module (pi, e, sqrt(), ceil(), floor(), pow(), fabs(), sin(), cos(), tan()); random module (random(), randint(), randrange()), statistics module (mean(), median(), mode()).Functions and its types (Built-in Functions, Functions defined in Module, User Defined Functions), arguments, default parameters, positional parameters, Function Returning Value(s), Recursion, Scope of a Variable.

Unit-4

Files: Introduction to files, types of files (Text file, Binary file, CSV file),Text file: opening a text file, file open modes (r, r+, w, w+, a, a+ etc), closing a text file, opening a file using with clause, writing/appending data to a text file using write() and writelines(), reading from a text file using read(), readline() and readlines

Reference Book: 1.The Complete Reference Python By Martin C Brown Publication by McGraw Hill. 2. Let us Python By Yashwant Kanetkar

B24-ESC-IT-215	Digital Electronics and Logic Design						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	**To provide the conceptual knowledge about the design of digital circuits**						
	Course Outcomes						
CO1	To introduce Simplification of switching functions using K map and QM methods						
CO2	To familiarize students with combinational circuit design						
CO3	Digital circuit design using sequential method						
CO4	To brief students how to change analog data into digital and vice versa.						

UNIT 1

Fundamentals of digital techniques: Review of logic gates and number system; 1's and 2's compliment Arithmetic ; Introduction to Boolean algebra using basic postulates and theorems ; Binary codes: BCD, Excess-3, Gray codes ; Standard representation of logic functions : SOP and POS forms; Simplification of switching functions using K map and Quine-McCluskey methods

UNIT 2

Design of Combinational circuits : Half and Full Adders ; Half and Full Subtractors ; Multiplexers and Demultiplexers / Decoders ; Implementation of SOP logic functions using multiplexers and Demultiplexers / Decoders; Encoders. Decoders / Drivers for display devices. , code converters

UNIT 3

Sequential circuits: Latches, Flip Flops: S-R- J-K. T, D, master-slave, edge triggered flip flop ;Race around condition; Excitation table ; Interconversion among flip flop, Design of Synchronous and Asynchronous counters ; Modulo N counter design ; Shift registers ;

UNIT 4

A/D and D/A converters: Sample and hold circuit, Quantization , weighted resistor and R -2 R ladder Digital to Analog Converters, Specifications for D/A converters., Flash type Analog to digital Converter ; Successive approximation type Analog to digital Converter , specifications of ADCs.

Programmable Logic Devices:

Introduction to PLA and PAL. , Implementation of simple functions using PLA and PAL

Suggested Books

- R. P. Jain , "Modem Digital Electronics (Edition III)" ; TMH
- Anand Kumar , "Fundamentals of digital circuits" ; PHI
- Malvino & Leach , "Digital Principles and Applications" , McGraw Hill.
- Thomas L. Floyd, "Digital Fundamentals" , Pearson Education Inc,

B24-IT-203	Data Structure						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
4	**0**	**0**	**4**	**70**	**30**	**100**	**3 Hour**
Purpose	**To introduce the principles and paradigms of Data Structures for design and implement the software systems logically and physically**						
	Course Outcomes						
CO1	To introduce the basic concepts of Data structure, basic data types, searching and sorting based on array data types.						
CO2	To introduce the structured data types like Stacks, Queue, and its basic operations' implementation.						
CO3	To introduces dynamic implementation of linked list.						
CO4	To introduce the concepts of Tree and graph and implementation of traversal algorithms.						

UNIT 1

Introduction to Data Structures: Definition & abstract data types, Real life applications with example; built in and user defined data structures.
Arrays: Definition, implementation, lower bound, upper bound, addressing an element at a particular index for one dimensional arrays, Two dimensional arrays and Multidimensional arrays. Implementation of Data Structures like structure, Sparse matrices: implementation of transpose.
Sorting & Searching : Basic Searching techniques (Linear & binary), Introduction to Sorting. Sorting using selection, insertion, bubble, merge, quick, radix, heap sort.

UNIT 2

Stacks : Sequential implementation of stacks, operations, Polish-notations, Evaluation of postfix expression, Converting Infix expression to Prefix and Postfix expression, Applications.
Queues: Definition, Sequential implementation of linear queues, Operations. Circular queue: implementation (using arrays), Advantage over linear queue, Priority queues & Applications.

UNIT 3

Linked Lists: Need of dynamic data structures, Operations on lists. Dynamic implementation of linked lists, Comparison between Array and Dynamic Implementation of linked list.Linked implementation of stacks and queues. Circular lists, implementation of primitive operations. Doubly linked lists: continuous & dynamic implementation, operations.

UNIT 4

Trees : Definition, Basic terminology, Binary tree, Array and Dynamic Implementation of a binary tree, primitive operations on binary trees. External and internal nodes. Binary tree traversals : preorder, inorder and postorder traversals. Representation of infix, postifix and prefix expressions using tree, Introduction to Binary Search Trees, B trees, B+ trees , AVL Trees, threaded trees, balanced multi way search trees.
Graphs: Definition of undirected & Directed Graphs & Networks, Basic terminology, Representation of graphs,. Graph traversals, minimum-spanning trees, computer representation of graphs.

Suggested Books:

- Tenenbaum , “Data Structures” , PHI Pub.
- Aho, Hopcroft , Ullman , “Data Structures and Algorithms” , Addison-Wesley.
- Horowitz & Sahni , “ Fundamentals of Data structures” , Addison-Wesley
- Robert Kruse , “Data Structures and Program Design” , PHI,
- Symour Lipschetz , “ Theory & Problems of Data Structures” , TMH

B24-IT-205	Object Oriented Programming Using C++						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
4	**0**	**0**	**4**	**70**	**30**	**100**	**3 Hour**
Purpose	**To introduce the principles and paradigms of OOPS for design and implementation of Object Oriented System**						
	Course Outcomes						
CO1	To introduce the basic concepts of object oriented programming language and the its representation						
CO2	To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.						
CO3	To introduce polymorphism, interface design and overloading of operator.						
CO4	To handle backup system using file, general purpose template and handling of raised exception during programming						

UNIT 1

Introduction to C++, C++ Standard Library, Basics of a Typical C++ Environment, Pre-processors Directives, Illustrative Simple C++ Programs. Header Files and Namespaces, libraryfiles. Concept of objects, basic of object modeling, object classes, associations, behaviors, description, Object Oriented Analysis & Object Modeling techniques,

Object Oriented Concepts: Introduction to Objects and Object Oriented Programming, Encapsulation (Information Hiding), Access Modifiers: Controlling access to a class, method, or variable(public, protected, private, package), Other Modifiers, Polymorphism: Overloading,, Inheritance, Overriding Methods, Abstract Classes, Reusability, Class's Behaviors.

Classes and Data Abstraction: Introduction, Structure Definitions, Accessing Members of Structures, Class Scope and Accessing Class Members, Separating Interface from Implementation, Controlling Access Function And Utility Functions, Initializing Class Objects: Constructors, Using Default Arguments With Constructors, Using Destructors, Classes : Const(Constant) Object And Const Member Functions, Object as Member of Classes, Friend Function and Friend Classes, Using This Pointer, Dynamic Memory Allocation with New and Delete, Static Class Members, Container Classes And Integrators, Proxy Classes, Function overloading.

UNIT 2

Operator Overloading: Introduction, Fundamentals of Operator Overloading, Restrictions On Operators Overloading, Operator Functions as Class Members vs. as Friend Functions,Overloading, <<, >> Overloading Unary Operators, Overloading Binary Operators.

Inheritance: Introduction, Inheritance: Base Classes And Derived Classes, Protected Members, Casting Base- Class Pointers to Derived- Class Pointers, Using Member Functions, OverridingBase –Class Members in a Derived Class, Public, Protected and Private Inheritance, Using Constructors and Destructors in derived Classes, Implicit Derived –Class Object To Base-Class ObjectConversion, Composition Vs. Inheritance.

UNIT 3

Virtual Functions and Polymorphism: Introduction to Virtual Functions, Abstract Base Classes And Concrete Classes, Polymorphism, New Classes And Dynamic Binding, Virtual Destructors, Polymorphism, Dynamic Binding.

Files and I/O Streams: Files and Streams, Creating a Sequential Access File, Reading Data From A Sequential Access File, Updating Sequential Access Files, Random Access Files,

Creating A Random Access File, Writing Data Randomly To a Random Access File, Reading Data Sequentially from a Random Access File. Stream Input/Output Classes and Objects, Stream Output, Stream Input, Unformatted I/O (with read and write).

UNIT 4

Templates & Exception Handling: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Parameters, Templates and Inheritance,Templates and Friends, Templates and Static Members. Introduction, Basics of C++ Exception Handling: Try Throw, Catch, Throwing an Exception, Catching an Exception, Re-throwing an Exception, Exception specifications, Processing Unexpected Exceptions, Constructors, Destructors and Exception Handling, Exceptions and Inheritance.

Suggested Books:

- Deitel , "C++ How to Program" , Prentice Hall
- Robert Lafore, "Object Oriented Programming in Turbo C++" , The Waite Group Press.
- Ravichandran , "Programming with C++" , 2003, TMH
- Balagurusamy , "Object oriented Programming with C++", Tata McGraw-Hill

B24-BSC-IT-217	PROBABILITY AND STATISTICS						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	To familiarize the prospective students with the fundamentals of probability & statistics and how to apply the principles to model and analyze various phenomena in fields like finance, economics, and engineering, aiding in making informed decisions and predicting outcomes.						
	Course Outcomes						
CO1	Introduce the fundamental concepts of probability to analyze and predict outcomes in real-life situations.						
CO2	Probability theory provides models of probability distributions (theoretical models of the observable reality involving chance effects) to be tested by statistical methods which has various engineering applications.						
CO3	Make the students familiar about basic statistics to analyze data sets using various measures of central tendency and dispersion.						
CO4	After completion of Unit IV, students will proficiently apply correlation and regression techniques, including calculating coefficients and determining lines of regression, to analyze relationships between variables in datasets.						

UNIT-I (10 Hrs)

Basic Probability: Introduction, additive law of probability, Conditional Probability, Independent Events, Bayes' Theorem. Random Variables: Discrete random variables, probability distribution, Probability mass function and distribution function, Expectation, Moments, Variance and standard deviation of discrete random variables.

UNIT-II (12hrs)

Continuous Probability distribution: Continuous random variables, probability distribution, Probability density function and distribution function, Expectation, Moments, Variance and standard deviation of Continuous random variables. Probability distributions: Binomial, Poisson and Normal - evaluation of statistical parameters for these three distributions.

UNIT-III (10hrs)

Basic Statistics: Measures of Central tendency: Mean, median, quartiles, mode, Geometric mean, Harmonic mean, Measures of dispersion: Range, Quartile deviation, mean deviation, standard deviation, coefficient of variation, Moments, Skewness and Kurtosis.

UNIT-IV (08hrs)

Correlation & Regression: Introduction,Correlation, Coefficient of correlation, methods of calculations, Lines of regression, Rank correlation.

Suggested Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons,2006.

2. P. G. Hoel, S. C. Port and C. J. Stone, Introduction to Probability Theory, Universal Book Stall, 2003 (Reprint).

3. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.

4. W. Feller, An Introduction to Probability Theory and its Applications, Vol. 1, 3rd Ed.,Wiley, 1968.

5. N.P. Bali and and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.

6. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.

7. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.

8.Veerarajan T., Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010.

B24-IT-207	Computer Organization & Architecture						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	Student will be able to understand the basic concepts of computer architecture and organization, and understand the key skills of constructing cost-effective computer systems.						
	Course Outcomes						
CO1	Be familiar with the Computer arithmetic and data representation						
CO2	Be familiar with the basic computer organization and design						
CO3	Be familiar with instruction set architecture and parallel processing.						
CO4	Be acquainted with the basic knowledge of I/O organization.						

UNIT- 1

Data representation and Computer arithmetic: Introduction to Computer Systems, Organization and Architecture, Von Neumann Architecture, evolution and computer generations; fixed point, Floating-point and Decimal arithmetic operations, Digital arithmetic algorithms for Addition, Subtraction, Multiplication using Booth's algorithm, Multiprocessors and Multicomputer, MIPS, MFLOPS. Memory Organization: Memory Hierarchy, Types of Memory, TLB

UNIT-2

Basic Computer organization and Design: General register organization, stack organization and common bus system, computer instructions, timing and control, Input, output and Interrupt: Interrupt cycle, Design drivers: common case, Amdahl's law.

Micro programmed Control organization, Control Memory, address sequencing, micro instruction format, Horizontal Vs Vertical micro-programming, design of control Unit, micro program sequencer, Hardwired v/s Micro-programmed. CISC and RISC: features and comparison.

UNIT-3

Instruction set Architecture: Instruction codes, instruction formats (Zero, One, Two and Three Address Instruction).Instruction cycle, reference instructions; Memory reference instructions.Various addressing modes. Pipeline and vector Processing, Parallel Processing, Flynn's Taxonomy, Pipelining, Instruction Pipeline, Basics of vector processing and Array Processors.

UNIT-4

Input-output organization: I/O interface. I/O Bus and interface modules, I/O versus Memory Bus. Asynchronous data transfer: Strobe control, Handshaking, Asynchronous serial transfer. Modes of Transfer: Programmed I/O, Interrupt driven I/O, Priority interrupt; Daisy chaining, Parallel Priority interrupt. Direct memory Access, DMA controller and transfer. Input output Processor, CPU-IOP communication, Serial communication.

Suggested Books:

● William Stallings, “Computer Organization and Architecture – Designing for Performance”, Sixth Edition, Pearson Education, 2003.

● Morris Mano, M., “Computer System Architecture,” 3/e, Pearson Education, 2005.

● John P. Hayes, “Computer Architecture and Organization,” 3/e, TMH, 1998.

● David A. Patterson and John L. Hennessy, “Computer Organization and Design: The Hardware/Software interface”, Third Edition, Elsevier, 2005.

● V.P. Heuring, H.F. Jordan, “Computer Systems Design and Architecture”, Second Edition, Pearson Education, 2004.

● Carl Hamacher, ZvonkoVranesic and SafwatZaky, “Computer Organization”, Fifth Edition, Tata McGraw Hill, 2002.

B24-IT-209	IT Workshop (Python) Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**2**	**1.0**	**60**	**40**	**100**	**3h**
Purpose	The course is designed to provide Basic knowledge of Python.						
	Course Outcomes						
CO1	To study fundamentals of python programming and implement basic programs.						
CO2	To implement the searching technique using python.						
CO3	To implement sorting techniques using python						
CO4	To implement matrix multiplication using python.						

LIST OF PROGRAMS

1. Write a program to compute the GCD of two numbers.
2. Write a program to find the square root of a number
3. Write a program to find the Exponentiation (power of a number)
4. Write a program to find the maximum of a list of numbers
5. Write a program for Linear search and Binary search
6. Write a program for Selection sort
7. Write a program for Insertion sort
8. Write a program to find first n prime numbers
9. Write a program to multiply matrices
10. Write a program that take command line arguments (word count) .
11. Write a program to find the most frequent words in a text read from a file.

B24-ESC-IT-219	Digital Electronics and logic design Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**2**	**1.0**	**60**	**40**	**100**	**3h**
Purpose	**To implement theoretical digital electronics into practical circuits**						
	Course Outcomes						
CO1	To verify the truth table for various gates.						
CO2	To Implement the Boolean Expression to design circuit for any function.						
CO3	To learn the various methods for counter design						
CO4	To design state machine circuits using sequential circuits.						

LIST OF EXPERIMENTS

1. Study of TTL gates – AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
2. Design & realize a given function using K-maps and verify its performance.
3. To verify the operation of multiplexer & Demultiplexer.
4. To verify the operation of comparator.
5. To verify the truth tables of S-R, J-K, T & D type flip flops.
6. To verify the operation of bi-directional shift register.
7. To design & verify the operation of 3-bit synchronous counter.
8. To design and verify the operation of synchronous UP/DOWN decade counter using J K flipflops & drive a seven-segment display using the same.
9. To design and verify the operation of asynchronous UP/DOWN decade counter using J K flipflops & drive a seven-segment display using the same.
10. To design & realize a sequence generator for a given sequence using J-K flip-flops.
11. Study of CMOS NAND & NOR gates and interfacing between TTL and CMOS gates.
12. Design a 4-bit shift-register and verify its operation.

Note : A student has to perform at least ten experiments.
Seven experiments should be performed from the above list. Remaining three experiments may either be performed from the above list or designed & set by the concerned institution as per the scope of the syllabus

B24-IT-211	Object oriented Programming Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**3**	**1.5**	**60**	**40**	**100**	**3h**
Purpose	**To design and implement the Object Oriented System**						
	Course Outcomes						
CO1	To familiarize with the class and objects						
CO2	To implement the concept of constructors						
CO3	To familiarize the concept of operator overloading						
CO4	To implement the concepts of Inheritance						

LIST OF EXPERIMENTS

1. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.

2. A point on the two numbers can represent dimensional plane: an X coordinate and a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:
Enter coordinates for P1: 3 4
Enter coordinates for P2: 5 7
Coordinates of P1 + P2 are : 8, 11

3. Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this.
Enter first number, operator, second number: 10/ 3
Answer = 3.333333
Do another (Y/ N)? Y
Enter first number, operator, second number 12 + 100
Answer = 112
Do another (Y/ N) ? N

4. A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that
uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this:
Enter your area code, exchange, and number: 415 555 1212
My number is (212) 767-8900

Your number is (415) 555-1212

5. Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results may be a DM object or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or metres and cenitmetres depending on the object on display.

6. Create a class rational which represents a numerical value by two double values-NUMERATOR & DENOMINATOR. Include the following public member Functions:

• constructor with no arguments (default).
• constructor with two arguments.
• void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator.
• Overload + operator to add two rational number.
• Overload >> operator to enable input through cin.
• Overload << operator to enable output through cout.

Write a main ( ) to test all the functions in the class.

7. Consider the following class definition

```
class father {
protected :int age;
public;
father (int x) {age = x;}
virtual void iam ( )
{ cout<< "I AM THE FATHER, my age is : "<< age<< end1:}
};
```

Derive the two classes son and daughter from the above class and for each, define iam ( ) to writeour similar but appropriate messages. You should also define suitable constructors for theseclasses.Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them.Declare pointer to father. Successively, assign addresses of objects of the two derived classes tothis pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism inaction.

8. Write a program that creates a binary file by reading the data for the students from the terminal.

`The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.

9. A hospital wants to create a database regarding its indoor patients. The information to store include

a) Name of the patient
b) Date of admission
c) Disease
d) Date of discharge

Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

NOTE : A student has to perform at least ten experiments.

Seven experiments should be performed from the above list. Remaining three experiments may either be performed from the above list or designed & set by the concerned institution as per the scope of the syllabus

B24-IT-213	Data Structure Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**3**	**1.5**	**60**	**40**	**100**	**3h**
Purpose	**To implement concepts of Data Structures.**						
	Course Outcomes						
CO1	To Implement searching and sorting based on array data types.						
CO2	To introduce the structured data types like Stacks and Queue and its basic operation's implementation.						
CO3	To introduces dynamic implementation of linked list.						
CO4	To introduce the concepts of Tree and implementation of traversal algorithms.						

LIST OF EXPERIMENTS

1. Write a program to search an element in a two-dimensional array using linear search.
2. Using iteration & recursion concepts write programs for finding the element in the array using Binary Search Method
3. Write a program to perform following operations on tables using functions only

a) Addition b) Subtraction c) Multiplication d) Transpose

4. Write a program to implement Queue.
5. Write a program to implement Stack.
6. Write a program to implement the various operations on string such as length of string concatenation, reverse of a string & copy of a string to another.
7. Write a program for swapping of two numbers using 'call by value' and 'call by reference strategies'.
8. Write a program to implement binary search tree. (Insertion and Deletion in Binary search Tree)
9. Write a program to create a linked list & perform operations such as insert, delete, update, reverse in the link list
10 .Write the program for implementation of a file and performing operations such as insert, delete, update a record in the file.
11. Create a linked list and perform the following operations on it

    a) Add a node b) Delete a node

12. Write a program to simulate the various searching & sorting algorithms and compare their timings for a list of 1000 elements.
13. Write a program to simulate the various graph traversing algorithms.
14 Write a program which simulates the various tree traversal algorithms.
15 Write a program to implement various Searching Techniques.
16 Write a program to implement Sorting Techniques.

B24-IT-202	Principles of Programming Languages						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
4	**0**	**0**	**4**	**70**	**30**	**100**	**3h**
Purpose	To introduce the principles and paradigms of programming languages for design and implement the software intensive systems.						
	Course Outcomes						
CO1	To introduce the basic concepts of programming language, the general problems and methods related to syntax and semantics.						
CO2	To introduce the structured data objects, subprograms and programmer defined data types.						
CO3	To outline the sequence control and data control						
CO4	To introduce the concepts of storage management using programming languages.						

UNIT-1

Introduction: Syntax and Semantics Introduction: A brief history, Characteristics of a good programming language, Programming language translators- compiler and interpreters, Elementary data types – data objects, variable and constants, data types. Specification and implementation of elementary data types, Declarations, type checking and type conversions, Assignment and initialization, Numeric data types, enumerations, Booleans and characters. Syntax and Semantics: Introduction, general problem of describing syntax, Formal method of describing Syntax, attribute grammar dynamic semantic.

UNIT-2

Structured data objects, Subprograms and Programmer Defined Data Types Structured data objects: Structured data objects and data types, specification and implementation of structured data types, Declaration and type checking of data structure, vector and arrays, records Character strings, variable size data structures, Union, pointer and programmer defined data objects, sets, files. Subprograms and Programmer Defined Data Types: Evolution of data type concept abstraction, encapsulation and information hiding, Subprograms, type definitions, abstract data types, over loaded subprograms, generic subprograms.

UNIT–3

Sequence Control and Data Control Sequence Control: Implicit and explicit sequence control, sequence control within expressions, sequence control within statement, Subprogram sequence control: simple call return, recursive subprograms, Exception and exception handlers, co routines, sequence control. Concurrency – subprogram level concurrency, synchronization through semaphores, monitors and message passing Data Control: Names and referencing environment, static and dynamic scope, block structure, Local data and local referencing environment, Shared data: dynamic and static scope, Parameter and parameter transmission schemes.

UNIT-4

Storage Management and Programming Languages Storage Management: Major run time elements requiring storage, programmer and system controlled storage management and phases, Static storage management, Stack based storage management, Heap storage management, variable

and fixed size elements. Programming Languages: Introduction to procedural, non-procedural, structured, logical, functional and object oriented programming language, Comparison of C and C++ programming languages.

Suggested Books:

• Terrence W. Pratt, Marvin V. Zelkowitz, Programming Languages Design and Implementation, Pearson. • Allen Tucker and Robert Noonan, Programming Languages–Principles and Paradigms, Tata McGraw-Hill,2009.

• Ellis Horowitz, Fundamentals of Programming Languages, Galgotia Publications, 2010.

• C. Ghezzi, Programming Languages Concepts, Wiley Publications, 2010.

B24-IT-204	Operating System						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	**To familiarize how an operating system controls the computer**						
	Course Outcomes						
CO1	To study about the process of Operating System and it's scheduling.						
CO2	To learn about interprocess communication and deadlocks.						
CO3	To learn about memory management and Virtual Memory.						
CO4	To learn about distributed system and file system of operating system.						

UNIT 1

Introductory Concepts: Operating System functions and characteristics, historical evolution of operating systems, Real time systems, Distributed systems, Methodologies for implementation of O/S service , system calls, system programs , interrupt mechanisms.

Processes: Processes model, process states, process hierarchies, implementation of processes, data structures used such as process table, PCB creation of processes, context switching, exit of processes. Process scheduling**:** objective, preemptive Vs non- preemptive scheduling, comparative assessment of different algorithms such as round robin, priority bases scheduling, FCFS, SJF, multiple queues with feedback.

UNIT 2

Interprocess communication: Race conditions, critical sections, problems of mutual exclusion, Peterson's solution, producer-consumer problem, semaphores, counters, monitors, message passing; Deadlocks: conditions, modeling, detection, recovery, avoidance, deadlock prevention.

UNIT 3

Memory Management: Multiprogramming with fixed partition, variable partitions, virtual partitions, virtual memory, paging, demand paging design and implementation issues in paging such as page tables, inverted page tables, page replacement algorithms, page fault handling, working set model, local vs global allocation, page size, segmentation and paging.

UNIT 4

File Systems: File type, attributes, access and security, file operations, directory structures, path names, directory operations, implementation of file systems, implementation of file and file operations calls, implementation of directories, sharing of files, disk space management, block allocation, free space management, logical file system, physical file system.

Distributed Systems: Introduction to H/W and S/W concepts in distributed systems, Network operating systems and NFS, NFS architecture and protocol, client- server model, distributed file systems, RPC- Basic operations, parameter passing, RPC semantics

Suggested Books :

- Peterson & Silberschatz , " Operating System concepts" Addison Wesley
- Brinch, Hansen, "Operating System Principles" PHI
- Tenanbaum " Operating System", PHI.

B24-IT-206	Advance Programming (Java)						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
4	**0**	**0**	**4**	**70**	**30**	**100**	**3 Hour**
Purpose	To introduce the concepts of advanced java and its implementations.						
	Course Outcomes						
CO1	Study fundamental concepts of Java.						
CO2	To study and implement String and Collection methods.						
CO3	To study oops concept and implement abstraction, inheritance						
CO4	To study servlet and data base connectivity of java and java servlets.						

UNIT – 1

Introduction: Importance and features of Java, Concepts of Java Virtual machine (JVM), Java Comments, Keywords, Constants, Variables and Data types, java Type Casting, Wrapper classes, Operators and Expressions, Control Statements, Conditional Statements, Loops and Iterations. Creating an Array of one and two dimensional . Java Math methods. Method parameter, Calling Methods.

UNIT – 2

String, String Buffer and string Builder classes. String methods (char At, concat, compare To, equals, get Chars, length, replace, to Lower Case, to Upper Case etc.).

Java Collection: list interface(Array, vector, stack, linked list),Queue interface, Map interface, Tree Set and Tree Map interface, Hash Map and Hash Set interface.

Exception Handling, Manual Exception creation, File Handling

UNIT – 3

Class definition, adding Variables and Methods, creating Objects, Constructors, java Modifier, java encapsulation, java inheritance, method Overloading and Over Riding, Java abstraction, Java interface, Java Packages.

UNIT – 4

JDBC: JDBC Fundamentals, Establishing Connectivity and working with connection interface, working with statements, Creating and Executing SQL statements(creation of table, insertion, deletion, updation).

Servlets: Introduction to Servlets, Life cycle of Servlets, Creating, Compiling and running Servlet, Reading the servlet Parameters, Reading Initialization parameter, Handling HTTP Request and Response (GET / POST Request), Session Tracking.

Suggested Books:

1. Gary Cornell and Horstmann Cay S., Core Java, Vol I and Vol II, Sun Microsystems Press.
2. Herbert Schildt, Java: The Complete Reference, McGraw-Hill.
3. Philip Hanna, JSP: The Complete Reference, McGraw-Hill.
4. Deital and Deital, Java How to Program, Prentice Hall (2007).

B24-IT-208	Data Base Management Systems						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
4	**0**	**0**	**4**	**70**	**30**	**100**	**3 Hour**
Purpose	**To familiarize the students with Data Base Management system**						
	Course Outcomes						
CO1	To provide introduction to relational model.						
CO2	To learn about ER diagrams and SQL.						
CO3	To understand about the concept of functional dependencies.						
CO4	To understand about Query Processing and Transaction Processing.						

UNIT I

Introduction : Concept & Overview of DBMS, Advantages of DBMS over file processing system, Database Languages, Responsibilities of Database Administrator, Database Users, Three Schema architecture of DBMS & Data Independence, Data Models.
Entity-Relationship Model: Basic concepts, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features: Specialization and Generalization.

UNIT 2

The Relational Data Model & Algebra : Relational Model: Structure of relational Databases, Relational Algebra & various operations(Set operation, select, project, joins, division), Relational Calculus: Domain , Tuple.
Integrity Constraints & Introduction To Sql:-
Domain Constraints, Referential Integrity Constraints, Basic Structure & Concept of DDL, DML, DCL, Aggregate Functions, Null Values, Introduction to views, Creating, modifying and deleting views.

UNIT 3

Relational Database Design : Functional Dependency, Different anomalies in designing a Database., Normalization – 1NF, 2NF, 3NF, Boyce-Codd Normal Form, Normalization using multivalued dependencies, 4NF, 5NF.

UNIT 4

Transaction Processing Concept: Introduction to transaction processing, transaction model properties, serializability:-Serial, non-serial and Serializable Schedules, Conflict Serializability.
Concurrency Control : Need of concurrency control, Different concurrency control Techniques: locking based, timestamps based technique. Deadlock handling and Recovery Techniques:- Deferred update/ immediate update, shadow paging.

Suggested Books:

- Elmasri and Navathe , “Fundamentals of Database Systems” , Addision-Wesley,
- Silberschatz, and Korth ,”Database System Concepts”, McGraw-Hill
- Date , “An Introduction to Database Systems” ,Addison-Wesley,
- Bhattacharyya, “Database Management Systems” , Tata McGraw-Hill Publishing.

B24-HSM-IT-216	Organizational Behavior						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**3**	**70**	**30**	**100**	**3 Hour**
Purpose	The objective of this Course is to make students conversant with the basic concepts of organization behavior for nurturing managerial skills.						
	Course Outcomes						
CO1	An overview about organizational behavior as a discipline and understanding the concept of individual behavior.						
CO2	Understand the concept and importance of personality, emotions and its importance in decision making and effective leadership.						
CO3	Enabling the students to know about the importance of effective motivation and its contribution in group dynamics and resolving conflicts.						
CO4	Understand how to overcome organizational stress by maintaining proper organizational culture and effective communication.						

UNIT- 1

Introduction to organizational behavior: Concept and importance of organizational behavior, role of Managers in OB, challenges and opportunities for OB.

Foundation of individual behavior: Biographical characteristics, concept and types of abilities , concept of values and attitude, types of attitude, attitude and workforce diversity.

UNIT- 2

Introduction to personality and emotions: Definition and Meaning of Personality, Determinants of Personality, Personality Traits Influencing OB, Nature and Meaning of Emotions, Emotions dimensions, concept of Emotional intelligence.

Perception and individual decision making: meaning of perception, factors influencing perception, rational decision making process, concept of bounded rationality. Leadership-trait approaches, behavioural approaches, situational approaches, and emerging approaches to leadership.

UNIT-3

Motivation: Concept and theories of motivation, theories of motivation-Maslow, two factor theory, theory X and Y, ERG Theory, McClelland's theory of needs, goal setting theory, application of theories in organizational scenario, linkage between MBO and goal setting theory.

Foundations of group behaviour and conflict management: Defining and classifying of groups, stages of group development, Informal and formal groups- group dynamics, managing conflict and negotiation , causes of group conflicts, managing intergroup conflict through resolution.

UNIT-4

Introduction to Organizational Communication: Meaning and importance of communication process, importance of effective communication, organizational stress:

definition and meaning sources and types of stress, impact of stress on organizations, stress management techniques.

Introduction to Organization Culture: Meaning and nature of organization culture, types of culture, managing cultural diversity, managing change and innovation-change at work, resistance to change, a model for managing organizational change.

Text Books:

1. Colquitt, Jason A., Jeffery A. LePine, and Michael Wesson. Organizational Behavior

: Improving Performance and Commitment in the Workplace. 5th ed. New York: McGrawHill Education, 2017.
2. Hitt, Michael A., C. Chet Miller, and Adrienne Colella. Organizational Behavior. 4th ed. Hoboken, NJ: John Wiley, 2015.
3. Robbins, Stephen P., and Timothy Judge. Organizational Behavior. 17th ed. Harlow, UK: Pearson Education, 2017. Stephen P. Robins, Organisational Behavior, PHI Learning / Pearson Education, 11th edition, 2008.

Reference Books:

1. Schermerhorn, Hunt and Osborn, Organisational behavior, John Wiley.
2. Udai Pareek, Understanding OrganisationalBehaviour, Oxford Higher Education.
3. Mc Shane & Von Glinov, OrganisationalBehaviour, Tata Mc Graw Hill.
4. Aswathappa, K., OrganisationalBehaviour– Text and Problem, Himalaya Publication.

B24-IT-210	Advanced Programming Lab (Java)						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**3**	**1.5**	**60**	**40**	**100**	**3Hour**
Purpose	To introduce the concepts of Advanced Java Programming						
	Course Outcomes						
CO1	Study fundamental concepts of Java.						
CO2	To study and implement String and Collection methods.						
CO3	To study oops concept and implement abstraction, inheritance						
CO4	To study servlet and data base connectivity of java and java servlets						

List of Practicals

1. Write a Java program to implement matrix multiplication

2. Write a java program to implement String, String Buffer and String builder.

3. Write a java program to implement Stack and queue.

4. Write a java program to handle File Exception, Arithmetic exception and Array out of bound Exception.

5. Write a java program to throw user defined exception.

6. Write a java program to implement multiple inheritance

7. Write a java program to calculate area of cube and volume using abstraction.

8. Write a java program to insert and display data from database.

9. Write a java servlet program to display data from HTML form.

10. Write a java servlet program to setup session tracking using cookie.

11. Write a java servlet program to set session tracking using hidden form and url rewriting.

12. Write a java servlet program to display data from employee table.

B24-IT-212	Operating System Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**2**	**1**	**60**	**40**	**100**	**3Hour**
Purpose	**To introduce the principles and paradigms of Operating System**						
	Course Outcomes						
CO1	To implement Process Scheduling algorithms.						
CO2	To implement deadlock.						
CO3	To implement Semaphores.						
CO4	To implement the program for memory allocation.						

LIST OF EXPERIMENTS

1. WAP to implement First Come First Scheduling (FCFS).
2. WAP to implement Shortest Job First Scheduling (SJF).
3. WAP to implement Priority Based Scheduling.
4. WAP to implement Banker's Algorithm.
5. WAP to implement LRU Page replacement Algorithm.
6. WAP to implement Round Robin Scheduling.
7. WAP to implement optimal page replacement algorithm.
8. WAP to implement producer-consumer problem.
9. WAP to implement first fit method.
10. WAP to implement best fit method.
11. WAP to implement worst fit method.
12. WAP to implement counting semaphores.

NOTE : A student has to perform at least ten experiments.
Seven experiments should be performed from the above list. Remaining three experiments may either be performed from the above list or designed & set by the concerned institution as per the scope of the syllabus

B24-IT-214	Database Management Systems Lab						
L	**T**	**P**	**Credit**	**Practical Exam**	**Internal Assessment**	**Total**	**Time**
0	**0**	**3**	**1.5**	**60**	**40**	**100**	**3Hour**
Purpose	**To implement practically the various concepts of DBMS**						
	Course Outcomes						
CO1	To understand & Implement basic DDL commands.						
CO2	To learn & Implement DML and DCL commands.						
CO3	To understand the SQL queries using SQL operators.						
CO4	To understand the concept of relational algebra and implement using examples.						

LIST OF EXPERIMENTS

1. Create a database and write the programs to carry out the following operation :
- Add , Delete and modify a record in the database
- Generate queries
- Data operations
- List all the records of database in ascending order.

2. To perform various integrity constraints on relational database.

3. Create a database and perform the following operations:-
- Arithmetic and Relational operations
- Group by & having clauses
- Like predicate for pattern matching in database

4. Create a view to display details of employees working on more than one project.

5. Create a view to display details of employees not working on any project.

6. Using two tables create a view which shall perform natural join, equi join, outer joins.

7. Write a procedure to give incentive to employees working on all projects. If no such employee found give app. Message.

8. Write a procedure for computing amount telephone bill on the basic of following conditions.
1. telephone rent Rs. 205 including first 105 free units.
2. if extra units>0 but <500 then rate is 80 paise per unit.
3. if extra units>500 then rate is Rs. 1.20 per unit.
For this purpose create a table with name, Phone No., No. of units consumed, bill amount of a customer.

9. Write a procedure for computing income tax of employee on the basic of following conditions:-
1. if gross pay<=40,000 then I.T rate is 0%.
2. if gross pay>40,000 but <60000 then I.T rate is 10%.
3. if gross pay>60,000 but <1,00,0000 then I.T rate is 20%.
4. if gross pay>1,00,0000 then I.T rate is 30%.
For this purpose create a table with name, ssn, gross salary and income tax of the employee.

10. Write trigger for before and after insertion, deletion and updation process.

NOTE : A student has to perform at least ten experiments.
Seven experiments should be performed from the above list. Remaining three experiments may either be performed from the above list or designed & set by the concerned institution as per the scope of the syllabus

B24-MC-IT-218	Environmental Sciences						
L	**T**	**P**	**Credit**	**End Semester Exam**	**Internal Assessment**	**Total**	**Time**
3	**0**	**0**	**1**	**70**	**30**	**100**	**3 Hour**
Purpose	To learn the multidisciplinary nature, scope and importance of Environmental sciences.						
CO1	The students will be able to learn the importance of natural resources.						
CO2	To learn the theoretical and practical aspects of eco system.						
CO3	Will be able to learn the basic concepts of conservation of biodiversity.						
CO4	The students will be able to understand the basic concept of sustainable development.						

UNIT 1

The multidisciplinary nature of environmental studies, Definition, Scope and Importance, Need for public awareness, Natural Resources: Renewable and Non-Renewable Resources: Natural resources and associated problems.

(a) Forest Resources: Use and over-exploitation, deforestation, case studies. Timber eztraction, mining, dams and their effects on forests and tribal people.

(b) Water Resources: Use & over-utilization of surface & ground water, floods, drought, conflicts over water, dams-benefits and problems.

(c) Mineral Resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.

(d) Food Resources: World Food Problems, changes caused by agriculture and overgazing, effects of modern agriculture, fertilizerpesticide problems, water logging, salinity, case studies.

(e) Energy Resources: Growing energy needs, renewable & non-renewable energy sources, use of alternate energy sources. Case studies.

(f) Land Resources: Land as a resource, land, degradation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources, Equitable use of resources for sustainable lifestyle.

UNIT II

Ecosystem-Concept of an ecosystem. Sturcture and function of an ecosystem, Producers, consumers and decomposers, Energy flow in the ecosystem, Ecological Succession, Food Chains, food webs and ecological pyramids. Introduction, types, characteristic features, structure and function of the following ecosystem: (a) Forest Ecosystem, (b) Grassland Ecosystem, (c) Desert Ecosystem and (d) Aquatic Ecosystems (ponds, streams, lakes, rivers, oceans, esturaries Field Work: Visit to a local area to document Environment assets-river/forest/grassland/hill/mountain, Visit to a local polluted site-Urban /Rural Industrial/Agricultural, Study of common plants, insects and birds, Study of simple ecosystems-pond, river, hill, slopes etc. (Field work equal to 5 lecture hours).

UNIT III

Biodiversity and its conservation: Introduction, Definition: genetic, species and ecosystem diversity. Biogeographical classification of India. Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. Biodiversityof global, National and local levels. India as a mega-diversity nation Hot spots of Biodiversity, Threats to biodiversity: Habitat loss, poaching of wild life, manwildlife conflicts, Endangered and endemic species of India, Conservation of Biodiversity- In situ and Ex-Situ conservation of biodiversity. Environmental Pollution Definition: Cause, effects and control measures of (a) Air Pollution (b) Water Pollution (c) Soil Pollution (d) Marine Pollution (e) Noise Pollution (f) Thermal Pollution (g) Nuclear Hazards Solid waste management- cause, effects and control measures of urban and industrial wastes, Role of an individual in prevention of pollution, Pollution case studies, Disaster management: floods, earthquake, cyclone and landslides

UNIT IV

Social Issues and the Environment. From unsustainable to sustainable development, Urban problems related to energy, Water conservation, rain water harvesting, watershed management. Resettlement and rehabilitation of people: Its problems and concerns, Case Studies: Environmental ethics-issues and possible solutions. Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case studies: Wasteland Reclamation, Consumerism and waste products, Environment Protection Act, Air (Prevention and Control of Pollution) Act, Water (Prevention and Control of Pollution) Act, Wildlife Protection Act, Forest Conservation Act, Issues involved in enforcement of environmental legislation, Public Awareness, Human population and the Environment, Population growth, variation among nations, Population explosion-Family Welfare Programme, Environment and human health. Human rights, Value Education, HIV/AIDS, Women and Child Welfare, Role of Information Technology in Environment and Human Health, Case Studies, Drugs and their effects; Useful and harmful drugs, Use and abuse of drugs, Stimulant and depressan drugs, Concept of drug deaddiction, Legal position on drugs and laws related to drugs.

Suggested Books

- Environmental Studies- Deswal and Deswal. Dhanpat Rai and Co
.• Environmental Science and Engineering Anandan, P. and Kumaravelan, R. 2009. Scitech Publications (India) Pvt. Ltd., India.
- Environmental Studies. Daniels Ranjit R. J. and Krishnaswamy. 2013. Wiley India.
- Environmental Science- Botkin and Keller. 2012. Wiley , India

Note: The Examiner will be given the question paper template to set the question paper.