INVERTIS UNIVERSITY BAREILLY

Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

Scheme of Instruction & Syllabi
of
Bachelor of Technology
(Computer Science and Engineering)

(With effect from academic session 2023-24)

(Dr. Gaurav Agarwal)
HOD CSE

(Prof. R. K. Shukla)
Dean Engineering & Technology

(Prof. Y.D.S. Arya)
Vice- Chancellor

Department of Computer Science and Engineering
INVERTIS UNIVERSITY
Invertis Village, Bareilly-Lucknow NH-24, Bareilly,
243123 U. P.

INVERTIS UNIVERSITY BAREILLY

Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

STUDY AND EVALUATION SCHEME
(With effect from academic session 2023-2024)
B.Tech. in Computer Science and Engineering
YEAR II, SEMESTER III

Sl. No.	Category	Course Code	Course Title/ Subjects	Hours per week			Evaluation Scheme		Total	Credits
				L	T	P	CA	EE		
THEORY										
1	Humanities & Social Sciences including Management Courses	HAS- 303	Industrial Psychology	2	0	0	15	35	50	2
2	Basic Science course	BAS-301	Mathematics-III	3	1	0	30	70	100	4
3	Professional Core Courses	BCS-301	Data Structures	2	1	0	25	50	75	3
4	Professional Core Courses	BCS-302	IT Infrastructure Management	2	1	0	25	50	75	3
5	Engineering Science Course	BCS-303	Digital Electronics	2	1	0	25	50	75	3
6	Professional Core Courses	BCS-304	Python	1	0	0	10	15	25	1
7	Engineering Science Course	IIOT3	Smart Industrial Connectivity	4	0	0	30	70	100	4
PRACTICALS AND PROJECTS										
7	Professional Core Courses	BCS-351	Data structures Lab	0	0	4	20	30	50	2
8	Professional Core Courses1	BCS-352	IT Infrastructure Management Lab	0	0	4	20	30	50	2
9	Engineering Science Course	BCS-353	Digital Electronics Lab	0	0	4	20	30	50	2
10	Professional Core Courses	BCS-354	Python Lab	0	0	4	20	30	50	2
			TOTAL	16	4	16	240	460	700	28

L-Lecture, **T**-Tutorial, **P**-Practical, **CA**-Continuous Assessment, **EE**-End Semester Examination, **HSMC**-Humanities & Social Science and Management Course

INVERTIS UNIVERSITY BAREILLY
Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

STUDY AND EVALUATION SCHEME
(With effect from academic session 2023-2024)
B.Tech. in Computer Science and Engineering
YEAR II, SEMESTER IV

Sl. No.	Category	Course Code	Course Title/ Subjects	Hours per week			Evaluation Scheme		Total	Credits
				L	T	P	CA	EE		
THEORY										
1	Humanities & Social Sciences including Management Courses	HAS-402	Industrial Sociology	2	0	0	15	35	50	2
2	Engineering Science Course	BCS-401	Computer Organization & Architecture	3	0	0	25	50	75	3
3	Mandatory Courses	BMC-001	Environmental Science	-	-	-		-	-	0
4	Professional Core Courses	BCS-402	Design and Analysis of Algorithms	3	0	0	25	50	75	3
5	Professional Core Courses	BCS-403	Operating Systems	3	0	0	25	50	75	3
6	Professional Core Courses	BCS-404	Software Engineering	3	0	0	25	50	75	3
7	Engineering Science Course	IIOT4	Data Analytics for IIOT	4	0	0	30	70	100	4
PRACTICALS AND PROJECTS										
7	Engineering Science Course	BCS- 451	Computer Organization & Architecture Lab	0	0	4	20	30	50	2
8	Professional Core Courses	BCS-452	Design and Analysis of Algorithms Lab	0	0	4	20	30	50	2
9	Professional Core Courses	BCS-453	Operating System Lab	0	0	4	20	30	50	2
			TOTAL	18	1	12	205	395	600	24

L-Lecture, **T**-Tutorial, **P**-Practical, **CA**-Continuous Assessment, **EE**-End Semester Examination, **HSMC**-Humanities & Social Science and Management Course, **MC**-Mandatory Course

HAS-303	Industrial Psychology	L T P 2 0 0	2 credits

Course Objectives:

CO1	Introduce major topics and subspecialties including critical theory and research finding that have defined the field of I/O psychology.
CO2	Increase the understanding of the complicated systems of individual and group psychological processes involved in the world of work
CO3	To connect the basic principles of I/O psychology to personnel and human resources management within the organization
CO4	Describe major topics and subspecialties including critical theory and research finding that have defined the field of I/O psychology
CO5	Describe the complicated systems of individual and group psychological processes involved in the world of work

MODULE-I

Introduction – Objectives and scope of Industrial Psychology, The Industrial Psychologist, Scientific management and Human Relations School – Hawthorne Experiments.

MODULE-II

Individual in Workplace -Motivation and Job satisfaction, stress management, Organizational culture, Leadership & group dynamics.

MODULE -III

Work Environment & Engineering Psychology-fatigue. Boredom, accidents and safety, Job Analysis, Recruitment and Selection – Reliability & Validity of recruitment tests, Performance Management - Training & Development.

Text Books:

1. Miner J.B. (1992) Industrial/Organizational Psychology. N Y: McGraw Hill.
2. Industrial psychology.S.N.chauhan, Sandeep Mittal,R.P.singh, Prateek Jain Pragati prakashan

Reference Books :

3. Blum & Naylor (1982) Industrial Psychology. Its Theoretical & Social Foundations CBS Publication.

COURSE OUTCOMES:

CO1	Aware about the field of I/O psychology.
CO2	Student can easily understand the complicated systems of individual and group psychological processes involved in the world of work
CO3	Psychology prepare for industry.
CO4	Able to find the research area in the field of psychology
CO5	Able to describe the complicated systems of individual and group psychological processes involved in the world of work

BAS-301	Mathematics III	L T P 3 1 0	4 credits

Course Objectives:

CO1	To understand the method of solving algebraic, transcendental equations.
CO2	determine the approximate value of the derivative & definite integral for a given data using numerical techniques.
CO3	to expand the given periodic function defined in the given range in terms of sine and cosine multiple of terms as a Fourier series.
CO4	to extreme the functional using integration technique.
CO5	know how root finding techniques can be used to solve practical engineering problems.

MODULE-I

Function of Complex variable: Analytic function, C-R equations, Cauchy's integral theorem, Cauchy's integral formula for derivatives of analytic function, Taylor's and Laurent's series, singularities, Residue theorem, Evaluation of real integrals.

MODULE-II

Statistical Techniques-I: Moments, Moment generating functions, Skewness, Kurtosis, Curve fitting, Method of least squares, Fitting of straight lines, Polynomials, Exponential curves etc., Correlation, Linear, non –linear and multiple regression analysis, Probability theory.
Statistical Techniques-II: Binomial, Poisson and Normal distributions, Sampling theory (small and large), Tests of significations: Chi-square test, t-test, Analysis of variance (one way) , Application to engineering, medicine, agriculture etc.
Time series and forecasting (moving and semi-averages), Statistical quality control methods, Control charts, , R, p, np, and c charts.

MODULE-III

Numerical Techniques–I: Zeroes of transcendental and polynomial equation using Bisection method, Regula-falsi method and Newton-Raphson method, Rate of convergence of above methods.
Interpolation: Finite differences, difference tables, Newton's forward and backward interpolation , Lagrange's and Newton's divided difference formula for unequal intervals.
Numerical Techniques–II: Solution of system of linear equations, Gauss- Seidal method, Crout method. Numerical differentiation, Numerical integration , Trapezoidal , Simpson's one third and three-eight rules, Solution of ordinary differential (first order, second order and simultaneous) equations by Euler's, Picard's and forth-order Runge- Kutta methods.

Text Books:

1. Jain, Iyenger& Jain, Numerical Methods for Scientific and Engineering Computation, New Age International, New Delhi , 2003.
2. Chandrika Prasad, Advanced Mathematics for Engineers, Prasad Mudralaya, 1996.
3. E. Kreysig, Advanced Engineering Mathematics, John Wiley & Sons, 2005.
4. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 2005.
5. Devi Prasad, An introduction to Numerical Analysis, Narosa Publication house, New Delhi 2006.

6. R.K. Jain & S.R.K. Iyenger, Advance Engineering Mathematics, Narosa Publication House, 2002.

Reference Books :

1. J.N. Kapur, Mathematical Statistics, S. Chand & company Ltd., 2000
2. Peter V. O'Neil, Advance Engineering Mathematics Thomson (Cengage) Learning

COURSE OUTCOMES:

CO1	Apply the Set theory and Relation concepts
CO2	Apply the Functions and define the recursive functions.
CO3	Apply Laplace transform to different applications
CO4	Apply Inverse Laplace transform to different applications.
CO5	Identify the permutations and combinations.

BCS-301	DATA STRUCTURES	L T P 2 1 0	3 credits

Course Objectives:

CO1	Introduce the concept of data structures through ADT including List, Stack, Queues
CO2	To understand concepts about searching and sorting techniques.
CO3	To understand basic concepts about stacks, queues, lists, trees and graphs.
CO4	Able to analyze algorithms and determine their time complexity
CO5	To enable them to write algorithms for solving problems with the help of fundamental data structures.

MODULE-I

Introduction: Basic Terminology, Elementary Data Organization, Algorithm**,** Time and space complexity of algorithms. Asymptotic notations, Abstract data types.
Arrays: Representation of arrays, insertion and deletion operations, Single and Multidimensional Arrays, Sparse Matrices and their representations. **Searching** : Sequential search, Binary Search, Comparison and Analysis
Linked List: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis. Garbage collection and memory compaction.

MODULE-II

Stacks: Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack. Application of stack: Prefix and Postfix Expressions conversion, Evaluation of postfix expression, Iteration and Recursion-Principles of recursion, Problem solving using recursion with examples such as binary search, Fibonacci numbers, and towers of Hanoi.
Queues: Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues, Dequeue and Priority Queue.

Graphs: Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List. Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm

MODULE-III

Trees : Basic terminology used with Binary Trees, Binary Tree Representation, Extended Binary Trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Constructing Binary Tree from given Tree Traversal, Binary Search Tree: Operation of Insertation , Deletion, Searching in Binary Search tree . Threaded Binary trees, Traversing Threaded Binary trees..
Concept & Basic Operations of AVL Tree. B Tree: definitions, algorithms and analysis.
Sorting: Bubble sort, Insertion Sort, Selection, Bubble Sort, Quick Sort, Two Way Merge Sort, Heap Sort.

Performance and Comparison among all the methods

Hashing: Hash Function, Collision Resolution Strategies

Text books and References:

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein "Data Structures Using C and C++" , PHI

2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publication

3. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGraw Hill

4. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education

5. Lipschutz, "Data Structures" Schaum's Outline Series, TMH

6. G A V Pai, "Data Structures and Algorithms", TMH

COURSE OUTCOMES:

CO1	For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness.
CO2	For a given Search problem (Linear Search and Binary Search) student will able to implement it.
CO3	For a given problem of Stacks, Queues and linked list student will able to implement it and analyze the same to determine the time and computation complexity.
CO4	Able to analyze algorithms and determine their time complexity
CO5	Student will able to implement Graph search and traversal algorithms and determine the time and computation complexity.

BCS-302	IT Infrastructure Management	L T P 2 1 0	3 credits

COURSE OBJECTIVES:

CO1	To understand underlying principles of IT infrastructure and management services.
CO2	To undersatnd IT systems, service delivery and service support process for providing a quality service.
CO3	To understand the basics of storage management
CO4	To study policies for security management and mitigate security related risks in the Organization
CO5	To understand the IT and cyber ethics and study cyber forensics law and cyber crimes.

MODULE-I

INTRODUCTION: Information Technology, Computer Hardware, Computer Software, Network and Internet, Computing Resources, Network topologies (bus, star, ring, mesh).
IT INFRASTRUCTURE: Design Issues, Requirements, IT System Management Process, Service Management Process, IT Infrastructure Library.
SERVICE DELIVERY PROCESS: Service Delivery Process, Service Level Management, Financial Management, Service Management, Capacity Management, Availability Management.

MODULE-II

SERVICE SUPPORT PROCESS: Service Support Process, Configuration Management, Incident Management, Problem Management, Change Management, Release Management.
STORAGE MANAGEMENT: Backup & Storage, Archive & Retrieve, Disaster Recovery, Space Management, Bare Machine Recovery, Data Retention.

MODULE-III

SECURITY MANAGEMENT: Security, Computer and internet Security, Physical Security, Identity Management, Access Management, Intrusion Detection.
IT ETHICS: Introduction to Cyber Ethics, Intellectual Property, Privacy and Law, Computer Forensics, Ethics and Internet, Cyber Crimes.
EMERGING TRENDS in IT: Electronics Commerce, Electronic Data Interchange, Bluetooth, Infrared.

Text Book*:*
1. Phalguni Gupta, Surya Prakash, Umarani Jayaraman, IT Infrastructure and its Management, TMH

COURSE OUTCOMES:

CO1	To describe basic IT infrastructure, storage management, security measures, cyber ethics, computer forensics, cyber laws and electronic commerce.
CO2	To summarize the design requirements for IT systems, service delivery and service support process for providing a quality service.
CO3	To relate various service delivery and service support process for development of a quality product.
CO4	To focus on various storage and security schemes to provide availability and safety of IT system.
CO5	To test the data collected at any cyber crime scene and organize it to find out the sequence of events responsible for present situation using computer forensic schemes.

BCS-303	DIGITAL ELECTRONICS	L T P 2 1 0	3 credits

Course Objectives:

CO1	Understand the concepts of various components.
CO2	Understand concepts that underpin the disciplines of analog and digital electronic logic circuits.
CO3	Understand various Number systems and Boolean algebra, the Boolean expression using Boolean algebra and design it using logic gates.
CO4	Understand Design and implementation of combinational circuits
CO5	Understand Design and develop sequential circuits.

MODULE-I

Digital system and binary numbers: Signed binary numbers, binary codes, cyclic codes, error detecting and correcting codes, hamming codes. Floating point representation Gate-level minimization: The map method up to five variable, don't care conditions, POS simplification, NAND and NOR implementation, Quine Mc-Clusky method (Tabular method).

MODULE-II

Combinational Logic: Combinational circuits, analysis procedure, design procedure, binary adder-subtractor, decimal adder, binary multiplier, magnitude comparator, decoders, encoders, multiplexers
Synchronous Sequential logic: Sequential circuits, storage elements: latches, flip flops, analysis of clocked sequential circuits, state reduction and assignments, design procedure. Registers and counters: Shift registers, ripple counter, synchronous counter, other counters.

MODULE-III

Memory and programmable logic: RAM, ROM, PLA, PAL.
Design at the register transfer level: ASMs, design example, design with multiplexers.
Asynchronous sequential logic: Analysis procedure, circuit with latches, design procedure, reduction of state and flow table, race free state assignment, hazards.

Text Book:

1. M. Morris Mano and M. D. Ciletti, "Digital Design", 4th Edition, Pearson Education

Reference Books :

1. Introduction to Digital Logic Design, JP Hayes, PHI.

2. The Art of Digital Design: An Introduction to Top-Down Design, Franklin P. Prosser, PHI.

COURSE OUTCOMES:

CO1	Understand the concepts of various components to design stable analog circuits.
CO2	Represent numbers and perform arithmetic operations.
CO3	Minimize the Boolean expression using Boolean algebra and design it using logic gates.
CO4	Analyze and design combinational circuit.
CO5	Design and develop sequential circuits.

BCS-304	Python	L T P 1 0 0	1 credits

Course Objectives:

CO1	Describe the core syntax and semantics of Python programming language.
CO2	Discover the need for working with the strings and functions.
CO3	Illustrate the process of structuring the data using lists, dictionaries, tuples and sets.
CO4	Infer the Object-oriented Programming concepts in Python.
CO5	Discuss concept of Function

Detailed Syllabus

MODULE-I

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type() Function and Is Operator, Dynamic and Strongly Typed Language

Control Flow Statements: The if Decision Control Flow Statement, The if…else Decision Control Flow Statement, The if…elif…else Decision Control Statement, Nested if Statement, The while Loop, The for Loop, The continue and break Statements, Catching Exceptions Using try and except Statement, , Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

MODULE-II

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Lifetime of Variables

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings,

MODULE-III

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, The del Statement.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, Polymorphism.

Text Books:

1. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018.

Reference Books:

1. Jake VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data", 1st Edition, O'Reilly Media, 2016.

2. Aurelien Geron, "Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", 2nd Edition, O'Reilly Media, 2019.

Course Outcomes:

CO1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
CO2	Express proficiency in the handling of strings and functions.
CO3	Identify the commonly used operations involving file systems and regular expressions.
CO4	To apply various types of Function
CO5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism

INVERTIS UNIVERSITY BAREILLY
Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

BCS-351	Data structures Lab	L T P 0 0 4	2 credits

List of Programs to be Implemented in C language

1. Implementation of Linear Search
2. Implementation of Binary Search.
3. Implementation of Largest and second largest in array
4. Implementation of Bubble Sort
5. Implementation of Selection Sort
6. Implementation of Insertion sort
7. Implementation of transpose of sparse matrix
8. Implementation of Single Linked List.
9. Implementation of Double Linked List
10. Implementation of Circular Linked List.
11. Implementation of Stack Using Single Linked List.
12. Implementation of Stack Using Array.
13. Implementation of Circular Queue Using Array.
14. Implementation of Queue Using Linked List.
15. Implementation of Tower of Hanoi
16. Implementation of Recursive Binary search
17. Implementation of Merge sort
18. Implementation of Quick Sort
19. Implementation of Heap sort
20. Implementation of Conversion of Infix Expression to Postfix Expression
21. Implementation of Postfix Expression Evaluation

BCS-352	IT Infrastructure Management Lab	L T P 0 0 4	2 credits

The following exercises should be done preferably on 'LINUX' platform

1	History and applications of Linux
2	Study the various variants of Linux (Linux Mint, Kali, Ubuntu, Fedora, etc.)
3	Creation of Login on server machine and steps to logon through client Machine
4	Installation of virtual machine on system (VMware)
5	Installing various operating systems on virtual machine
6	Basic commands of Linux
7	Creation of files and merging of files using cat command
8	Vi Editor basics
9	Create a text file containing records of students in text form line wise, save this file and edit its copy
10	Understanding the "diff" command
11	Use "diff "command to generate a patch for the original file
12	Use the "patch" command to patch the original file to make it similar to the edited copy

BCS-353	DIGITAL ELECTRONICS LAB	L T P 0 0 4	2 credits

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 flip-flops & drive a seven-segment display using the same.
9. To design and verify the operation of asynchronous UP/DOWN decade counter using J K flip-flops & drive a seven-segment display using the same.
10. To design & realize a sequence generator for a given sequence using J-K flip-flops.

INVERTIS UNIVERSITY BAREILLY
Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

BCS-354	Python Lab	L T P 0 0 4	2 credits

List of Programs:

S.No	List of Programs
1	Write a program to demonstrate different number datatypes in python.
2	Write a program to perform different arithmetic operations on numbers in python
3	Create a list and perform the following methods 1) insert() 2) remove() 3) append() 4) len() 5) pop() 6) clear()
4	Create a dictionary and apply the following methods 1) Print the dictionary items 2) access items 3) use get() 4)change values 5) use len()
5	C) Create a tuple and perform the following methods 1) Add items 2) len() 3) check for item in tuple 4)Access iems
6	Write a python program to add two numbers.
7	Write a python program to print a number is positive/negative using if-else.
8	Write a python program to find largest number among three numbers.
9	Write a python Program to read a number and display corresponding day using if_elif_else?
10	Write a program to create a menu with the following options 1. TO PERFORM ADDITITON 2. TO PERFORM SUBTRACTION 3. TO PERFORM MULTIPICATION 4. TO PERFORM DIVISION Accepts users input and perform the operation accordingly. Use functions with arguments.
11	Write a python program to check whether the given string is palindrome or not.
12	Write a python program to find factorial of a given number using functions
13	Write a Python function that takes two lists and returns True if they are equal otherwise false
14	Write a program to double a given number and add two numbers using lambda()?
15	Write a program for filter() to filter only even numbers from a given list.

HAS-402	INDUSTRIAL SOCIOLOGY	L T P 2 0 0	2 credits

COURSE OBJECTIVES:

CO1	To provide you with an understanding of the ways in which the process of industrialization has shaped societies.
CO2	To understand the influence of the wider societal context on the operations within their organizations
CO3	Understand the role of evidence in the social sciences and the application of systematic empirical inquiry
CO4	Obtain sociological knowledge of core areas and substantive topics and the ability to think critically about them
CO5	Understand the role of theory in the application of conceptual frameworks in the research process

MODULE - I

Industrial Sociology:
Nature and Scope of Industrial Sociology, Development of Industrial Sociology, Rise and Development of Industry, Early Industrialism Types of Productive Systems, The Manorial or Feudal system, The guild system, The domestic or putting-out system and the factory system, Characteristics of the factory system

MODULE - II

Industrialization :
Causes and consequences of industrialization.
Industrialization in India.
Industrial Poling Resolutions – 1956.

MODULE - III

Contemporary Issues :
Grievances and Grievance handling Procedure.
Industrial Disputes : courses, strikes & lockouts,
Industrial Relations Machinery Bi-partite Tri-partite Agreement, Labour courts &Industrial Tribunals, Code of Discipline, Standing order.

Course Outcomes:

CO1	Aware about the major social groups that function in society, including racial and ethnic groups.
CO2	Aware about Development of Industrial Sociology
CO3	Knowledge about Industrialization in India.
CO4	Able to find the research area in the field of Sociology
CO5	Aware about Contemporary Issues in the field of Industrial Sociology

INVERTIS UNIVERSITY BAREILLY
Established by Govt. of U.P. u/s 2f of UGC Act, 1956 vide U.P. Act 22 of 2010.

BCS-401	Computer Organization & Architecture	L T P 3 0 0	3 credits

Course Objectives:

CO1	Conceptualize the basics of organizational and architectural issues of a digital computer.
CO2	Understand concepts of register transfer logic and arithmetic operations.
CO3	Understand the various computer architectures and control units
CO4	How I/O devices are accessed and its principles.
CO5	The current state of art in memory system design

MODULE-I

Introduction:, Digital computer Block diagram, functional units and their interconnections, buses, types of buses and bus arbitration.

Number representation : Fixed point Integer representation , Fixed point arithmetic operations in 2's complement form: Addition, Subtraction, Booths multiplication algorithm, array multiplier, and Division techniques. Floating point number representation, IEEE standard for floating point representation, Floating point arithmetic operation .

MODULE-II

Central Processing unit: Register, bus and Memory transfer, Register Transfer language, Arithmetic, logic and shift micro operations, arithmetic and logic unit, Processor organization: Single Accumulator, general register and stack organization, Addressing modes, Instruction types, Instruction formats, instruction cycle.

Control Unit :Hardwired and microprogrammed control, concept of horizontal and vertical microprogramming.

MODULE-III

Memory: Basic concept and hierarchy, semiconductor RAM memories, ROM memories, Cache memory, address mapping techniques and replacement, Auxiliary memories, Virtual memory.

Input / Output: Peripheral devices, I/O interface, I/O ports, Interrupts, types of interrupts, Modes of Data Transfer, Programmed I/O, interrupt initiated I/O and Direct Memory Access.
Pipelining: Basic concepts of pipelining, throughput and speedup.

Text Books:

1. William Stalling- Computer Organization and Architecture,PHI
2. Morris Mano, Computer System Architecture, PHI

Reference Books:

1. "Computer Organization and Design: The Hardware/Software Interface" , 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.
2. John P. Hays – Computer Organization , Mc-Graw Hill
3. Vravice,Hamacher & Zaky, "Computer Organization", TMH

Course Outcomes:

CO1	Draw the functional block diagram of a single bus architecture of a computer and describe the function of the instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set.
CO2	Define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
CO3	Understand the architecture and functionality of central processing unit.
CO4	Exemplify in a better way the I/O and memory organization.
CO5	Given a CPU organization and instruction, design a memory module and analyze its operation by interfacing with the CPU.

BMC-001	Environmental Science	L T P 0 0 0	Credits 0

Course Objectives:

CO1	To provide understanding of component of environment, their function ,quality, issues related to environment ,effect of quality degradation on human beings and their solutions.
CO2	To impart students with strong knowledge base through theory courses and sessional that makes them suitable for industries, academics, research and consultancies.
CO3	To develop students analytical, computational and research skills through assignments, weekly presentations and modeling software.
CO4	To train the students on developing practical, efficient and cost effective solutions on problems and challenges on environmental sciences and engineering.
CO5	To inculcate among students sensitivity towards social and corporate responsibilities.

MODULE 1

Introduction to Environmental Science - Definition and scope and need for public awareness Ecosystems, Concept, structure and functions, restoration of damaged ecosystems Biodiversity – Definition, description at national and global level, threats and conservation

MODULE 2

Natural Resources - Renewable and non-renewable and their equitable use for sustainability, Material cycles – carbon, nitrogen and sulphur cycle. Conventional and Non-conventional Energy Sources – fossil fuel-based, hydroelectric, wind, -nuclear and solar energy, biomass, biodiesel, hydrogen as an alternative fuel.

MODULE 3

Transportation and industrial growth Social Issues Related to Environment–Sustainable development, reset lement and rehabilitation Environmental ethics.

MODULE 4

Environmental Changes and Human Health Environmental Pollution–Definition, causes and effects, control measures for water, air, soil, noise, thermal pollution.

Textbook:

1. Environmental Studies, J Krishna wamy, RJ Ranjit Daniels, Wiley India.

Reference Books:

2. Environmental Science, Bernard J. Nebel, Richard T. Right, 9780132854467, Prentice Hall Professional 1993.
3. Environment and Ecology, RK Khandal, 978-81-265-4277-2, Wiley India.

Course OutCome:

CO1	To develop environmental scientists and engineers and sensitize them towards environmental issues.
CO2	To acquire analytical skills in assessing environmental impacts through a multidisciplinary approach.
CO3	To identify environmental problems and solutions through organized research.
CO4	To improve the communication and writing skill so as to face the competitive world
CO5	understanding of different component of environment and their function and sustainable Development

BCS-402	Design and Analysis of Algorithms	L T P 3 0 0	3 credits

Course Objectives:

CO1	To understand and remember algorithms and its analysis procedure.
CO2	To introduce the concept of data structures through ADT including List, Stack, Queues. Analyze the asymptotic performance of algorithms.
CO3	To design and implement various data structures and algorithms.
CO4	Apply important algorithmic design paradigms and methods of analysis.
CO5	Synthesize efficient algorithms in common engineering design situations.

MODULE-I

Introduction: Algorithms, Analyzing algorithms, Complexity of algorithms, Growth of Functions- master theorem, Sorting- insertion sort, selection sort, Shell sort, Comparison of sorting algorithms, sorting in linear time.
Divide and Conquer: General method, merge sort, quick sort, Heap sort, Strassen's matrix multiplication algorithm.
Advanced data Structures: B – trees , Red-Black trees, Data Structure for Disjoint sets.

MODULE-II

Graphs: Minimum Spanning trees – Prim's and Kruskal's algorithms, Single source shortest paths - Dijkstra's and Bellman Ford algorithms, Multistage graphs, all pairs shortest paths: Warshal's and Floyd's algorithms
The Greedy Method : optimal storage on tapes, Fractional Knapsack problem, Job sequencing with deadlines,.
Dynamic Programming: Introduction, O/1 knapsack,. Matrix chain multiplication, longest Common sequence.

MODULE-III

Back Tracking: Introduction, 8 queen's problem, graph coloring, Hamiltonian cycles, Subset Sum Problem.
Branch and Bound: Introduction, traveling salesperson problem, Euclids algorithm for GCD
String matching: Naïve String Matching, Rabin karp, Knuth-Morris-Pratt algorithm.
NP Completeness : Introduction to P, NP hard and NP completeness, NP complete problems : Clique problem, Vertex Cover problem, Travelling salesman problem.

Text Books:
1. Fundamental of Computer algorithms, Ellis Horowitz and Sartaj Sahni, 1978, Galgotia Publications
2. Introduction To Algorithms, Thomas H Cormen, Charles E Leiserson And Ronald L Rivest: , TMH

Reference Books:

1. The Design and Analysis of Computer Algorithm, Aho A.V. Hopcroft J.E., 1974, Addison Wesley.
2. Algorithms-The Construction, Proof and Analysis of Programs, Berlion, P.Bizard, P., 1986. Johan Wiley & Son
3. Introduction to Design and Analysis of Algorithm, Goodman, S.E. & Hedetnieni, 1997, MGH.

4. Introduction to Computers Science- An algorithms approach , Jean Paul Trembley, Richard B.Bunt.

Course Outcomes:

CO1	For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms .
CO2	Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms.
CO3	Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Synthesize divide-and-conquer algorithms. Derive and solve recurrence relation.
CO4	Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it
CO5	For a given model engineering problem, model it using graph and write the corresponding algorithm to solve the problems.

BCS-403	Operating Systems	L T P 3 0 0	3 credits

Course Objectives:

CO1	To understand the main components of an OS & their functions.
CO2	To study the process management and scheduling.
CO3	To understand various issues in Inter Process Communication (IPC) and the role of OS in IPC.
CO4	To understand the concepts and implementation Memory management policies and virtual memory.
CO5	To understand the concept of file organization and access mechanism.

MODULE-I

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching.
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads.
Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, **Scheduling criteria:** CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling.

MODULE-II

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Dekker's solution, Peterson's Solution, The Producer/Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.
Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

MODULE-III

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition– Internal and External fragmentation and Compaction; Paging: Page fault, Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).
Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference
I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers.
File Management: Concept of File, Access methods, File types, File operation, Directory structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping).
Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.

Text Books:

1. Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley
2. Sibsankar Halder and Alex A Aravind, "Operating Systems", Pearson Education
3. Harvey M Dietel, " An Introduction to Operating System", Pearson Education
4. D M Dhamdhere, "Operating Systems : A Concept based Approach", 2nd Edition, 13 TMH
5. William Stallings, "Operating Systems: Internals and Design Principles", 6th Edition, Pearson Education.

Course Outcomes:

CO1	Describe the important computer system resources and the role of operating system in their management policies and algorithms.
CO2	Understand the process management policies and scheduling of processes by CPU.
CO3	Evaluate the requirement for process synchronization and coordination handled by operating system.
CO4	Describe and analyze the memory management and its allocation policies.
CO5	Set file access permissions and protect and secure files.

BCS-404	SOFTWARE ENGINEERING	L T P 3 1 0	4 credits

Pre-requisites: Basic computer knowledge

Course Objectives:

CO1	to apply engineering and computer science concepts in the development and maintenance of reliable, usable, and dependable software
CO2	To understand the nature of software development and software life cycle process models, agile software development.
CO3	To learn concepts and principles in parallel with the software development life cycle.
CO4	To know basics of testing and understanding concept of software quality assurance and software configuration management process.
CO5	To understand project scheduling concept and risk management associated to various type of projects.

Detailed Syllabus

MODULE-I

Introduction: Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, and Software Quality Attributes. Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Model. Software Requirement Specifications (SRS) Requirement Engineering Process: Elicitation, Analysis, Documentation and Review, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.

MODULE-II

Software Design: Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, cyclomatic Complexity Measures: Control Flow Graphs. Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and BottomUp Testing Strategies: Test Drivers and Test Stubs, White Box Testing, Black Box Testing, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.

MODULE-III

Software Maintenance and Software Project Management: Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re- Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.

Text Books:-

1. R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. Rajib Mall, Fundamentals of Software Engineering, PHI Publication.

Reference Books:-

1. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.
2. PankajJalote, Software Engineering, Wiley
3. Carlo Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.

Course Outcomes: After the completion of the course the student will be able to:

CO1	To understand basic concept of software engineering, different phases to make a software & study them in detail, project management concepts & their metrics, design models & its principles
CO2	Discuss requirement engineering and its models (Information, functional, behavioural), different testing techniques for different projects
CO3	Implement Software life cycle models
CO4	Compare different types of models
CO5	Calculation of staffing for a particular project, its cost & schedule

BCS-451	Computer Organization & Architecture Lab	L T P 0 0 4	2 credits

List Of Practicals:

1. Implementation of Flip-Flops: SR, JK, D, T, Master Slave
2. Implementation of counters , up and Down Counters
3. Implementation of shift registers SISO, SIPO, PISO, PIPO
4. Implementation of Binary Adder.
5. Implementation of Binary Subtractor
6. Implementation of Seven Segment Display.

BCS-452	Design and Analysis of Algorithms Lab	L T P 0 0 4	2 credits

List of Programs:

1 Program for Quick Sort
2 Program for Merge Sort
3 Program for Heap sort
4 Program for Insertion sort
5 Program for counting sort
6. Program for Radix sort
7 Program for Knapsack Problem
8 Program for prims algorithm
9 Program for Kruskals Algorithm
10 Program for k-th element to find minimum and maximum
11. Program for Warshal's Algorithm
12. Program for Dijkstra's Algorithm

BCS-453	Operating System Lab	L T P 0 0 4	2 credits

List of Programs

1. Write a program in C to implement FCFS cpu scheduling.
2. Write a program in C to implement SJF cpu scheduling.
3. Write a program in C to implement Round robin cpu scheduling.
4. Write a program in C to implement Priority cpu scheduling.
5. Write a program in C to implement FIFO page replacement algorithm.
6. Write a program in C to implement LRU page replacement algorithm.
7. Write a program in C to implement OPTIMAL page replacement algorithm.
8. Simulate Banker's Algorithm for Dead Lock Avoidance.
9. Simulate Banker's Algorithm for Dead Lock Prevention.
10. Write a program in C to simulate FCFS disk scheduling algorithms
11. Write a program in C to simulate SCAN disk scheduling algorithms
12. Write a program in C to simulate C-SCAN disk scheduling algorithms