INSTITUTE OF ENGINEERING & TECHNOLOGY

SITAPUR ROAD, LUCKNOW

Evaluation Scheme & Syllabus

For

B. Tech. Second Year

Computer Science & Engineering

Computer Science & Engineering (AI)

(Effective from session 2023 – 24)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Structure, Evaluation Scheme and Syllabus

SEMESTER - III

Sl. No.	Course Code	Subject	Type	Periods			Evaluation Scheme				ESE		Total	Credit
				L	T	P	CT	TA	Total	PS	TE	PE		
1	*IOE 3XX /IAS303	Interdepartmental Courses /Math-III	ES / BS	3	1	0	20	10	30	-	70	-	100	4
2	IAS301 / IVE301	Technical Communication / Universal Human Values	HS / VE	3	0	0	20	10	30	-	70	-	100	3
3	ICS301	Data Structures	PC	3	1	0	20	10	30	-	70	-	100	4
4	ICS302	Digital Logic and Computer Organization	PC	3	1	0	20	10	30	-	70	-	100	4
5	ICS303	Discrete Structure and Theory of Logic	PC	3	1	0	20	10	30	-	70	-	100	4
6	ICS351	Data Structures using C Lab	PL	0	0	2	-	-	-	50	-	50	100	1
7	ICS352	Digital Logic and Computer Organization Lab	PL	0	0	2	-	-	-	50	-	50	100	1
8	ICS353	IT Tools (Linux Shell programming and LATEX)	PL	0	0	2	-	-	-	50	-	50	100	1
9	ICS354	Mini Project or Internship Assessment	PL	0	0	2	-	-	-	50	-	50	100	1
10	INC301 / INC302	Python Programming / Computer System Security	VA	3	0	0	20	10	30	-	70	-	100	0
		Total		**18**	**4**	**8**							**900**	**23**

SEMESTER – IV

Sl. No.	Course Code	Subject	Type	Periods			Evaluation Scheme				ESE		Total	Credit
				L	T	P	CT	TA	Total	PS	TE	PE		
1	IOE 4XX/ IAS403	Interdepartmental Courses /Math-III	ES / BS	3	1	0	20	10	30	-	70	-	100	4
2	IAS401 /IVE401	Technical Communication /Universal Human Values	HS / VE	3	0	0	20	1-0	30	-	70	-	100	3
3	ICS401	Operating System	PC	3	1	0	20	10	30	-	70	-	100	4
4	ICS402	Object Oriented Programming using JAVA	PC	3	1	0	20	10	30	-	70	-	100	4
5	ICS403	Theory of Automata and Formal Languages	PC	3	1	0	20	10	30	-	70	-	100	4
6	ICS451	Operating System Lab	PL	0	0	2	-	-	-	50	-	50	100	1
7	ICS452	Object Oriented Programming using JAVA Lab	PL	0	0	2	-	-	-	50	-	50	100	1
8	ICS453	Python Programming Lab	PL	0	0	2	-	-	-	50	-	50	100	1
9	ICS454	Web Technology based Mini Project	PL	0	0	2	-	-	-	50	-	50	100	1
10	INC401 /INC402	Python Programming /Computer System Security	VA	3	0	0	20	10	30	-	70	-	100	0
		Total		**18**	**4**	**8**							**900**	**23**

Open Electives

1. Basics of Data Structures and Algorithms (For all except CS Major and CS Minor students)
2. Computer Based Numerical Techniques (For all branches excluding CE, if covered in their Math Course)

*****IOE 3XX Subject code to be centrally decided

SYLLABI

ICS 301: Data Structures

Course Outcome (CO)		Bloom's Level
At the end of course , the student will be able to understand		
CO 1	Understand linear data structures, their implementation and application	K_3, K_4
CO 2	Understand stack and queue and demonstrate the knowledge in the application to the problems involving recursion	K_3, K_4
CO 3	Implementation of binary tree, tree operations including traversal.	K_2, K_3
CO 4	Understanding Graph, graph operations, implementation and applications.	K_2, K_3
CO 5	Understand and analyze the complexity of various sorting algorithms	K_3, K_4

DETAILED SYLLABUS

Unit	Topic	Lecture
I	**Introduction:** Basic Terminology, Elementary Data Organization, Built in Data Types in C. Algorithm, Time and Space Complexity, Asymptotic notations: Big Oh, Big Theta and Big Omega, Time-Space trade-off. Abstract Data Types (ADT) **Arrays**: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D,2-D,3-D and n-D Array Application of arrays, Sparse Matrices and their representations. **Linked lists:** Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List. Operations on a Linked List: Insertion, Deletion, Traversal. Polynomial Representation, Addition, Subtraction, Multiplications of Single variable and two variables Polynomial.	08
II	**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Tail recursion, Removal of recursion, Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Tradeoffs between iteration and recursion. **Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.	08
III	**Trees:** Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Array Representation and Pointer(Linked List) Representation, Binary Search Tree, Strictly Binary Tree ,Complete Binary Tree and Extended Binary Trees. Tree Traversal algorithms: Inorder, Preorder and Post order, Binary search tree: Creation, Insertion, Deletion, and Searching. Threaded Binary trees. Huffman coding using Binary Tree. Concept & Basic Operations for AVL Tree , B Tree & Binary Heaps	08
IV	**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. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm.	08
V	**Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing. **Sorting:** Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort and Radix Sort.	08

Text books:

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI , India
2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.

Reference books:

1. Thareja, "Data Structure Using C" Oxford Higher Education.
2. Michael T. Goodrich, Roberto Tamassia, David M. Mount "Data Structures and Algorithms in C++", Wiley India.
3. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education.
4. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGraw Hill.
5. Adam Drozdek "Data Structures and Algorithm in Java", Cengage Learning

ICS 302: Digital Logic and Computer Organization

Course Outcome (CO)		Bloom's Level
At the end of course, the student will be able to		
CO 1	Understand binary number system, Boolean algebra and design simple combinational circuits	**K_2, K_3**
CO 2	Understand basic flip flops and design simple sequential circuits	**K_2, K_3**
CO 3	Describe the fundamental organization of a computer system and understand computer arithmetic	**K_2, K_3**
CO 4	Demonstrate the knowledge of the functional units of a processor, addressing modes, instruction formats and program control statements	**K_2, K_3**
CO 5	Understand memory hierarchy, Cache mapping and I/O organization	**K_2**

DETAILED SYLLABUS		
Unit	**Topic**	**Lecture**
I	Number System: Binary, Octal, Hexadecimal System, Number Base Conversion, Digital Logic Gates, Boolean Algebra, SOP and POS forms, , Theorems and Properties of Boolean algebra, Gate Level Minimization: Karnaugh Map, Don't Care Condition, Combinational Logic Circuits: Half Adder, Full Adder, decoder, Encoder, Multiplexer	08
II	Sequential Circuits, Flip-Flops: RS, SR, JK, D, T, Clocked Sequential Circuits, Master-Slave flip-flop, Shift Registers, Ripple Counters, Synchronous Counters, Counter Design, Code Converters: Gray, BCD	08
III	Basic Functional units of Computers: Functional units, basic Operational concepts, Bus structures. Data Representation: Signed number representation, fixed and floating point Representations. Computer Arithmetic: Addition and subtraction with signed magnitude, Multiplication algorithm, Booth's multiplication, Array multiplier, Division algorithm: restoring and non-restoring division, Floating point arithmetic. Register Transfer Language and Micro Operations: RTL- Registers, Register transfers, Bus and memory transfers. Micro operations: Arithmetic, Logic and Shift micro operations Basic Computer Organization and Design: Computer Registers, Computer instructions, Instruction cycle. Instruction codes, Timing and Control, Types of Instructions: Memory Reference Instructions,	08
IV	CPU Organization: General Register Organization, Stack organization, Instruction formats: Three-Address, Two-Address, One- Address, and Zero-Address Instructions, Instruction cycle, Addressing modes, Data Transfer and Manipulation, Program Control, CISC and RISC processors Characteristics. Control Unit: Design approaches, Hardwired Control, Control memory, Address sequencing, micro program example, Micro Programmed Control.	08
V	Memory Organization: Memory hierarchy, Interleaving, Primary and Auxilary Memory, Associative memory. Cache Mapping Techniques: Direct, Associative and Set-Associative Mapping Input –Output Organization: Input-output subsystems, I/O device interface, I/O Processor, Modes of transfer, Interrupt, Priority Interrupt, I/O Transfers–Program controlled, Interrupt driven, and DMA, Pipelining	08

Text books:

1. M. Morris Mano, "Computer System Architecture", PHI
2. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, "Computer Organization", Tata McGraw Hill, Fifth Edition, 2002

Reference books:

1. M. Morris Mano, "Digital Logic and Computer Design", PHI
2. William Stallings, "Computer Organization and Architecture – Designing for Performance", Pearson Education, Ninth Edition, 2012
3. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software interface", Elsevier, 3rd Edition, 2005
4. John P. Hayes, "Computer Architecture and Organization", Tata McGraw Hill, Third illustrated Edition, 2007.

ICS 303: Discrete Structures and Theory of Logic

Discrete Structures & Theory of Logic (ICS 303)		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to understand		
CO 1	Understand the concept of relations, functions and apply the method of proof by Induction to the problems	K_3, K_4
CO2	Apply the inference theory of logic in logical problems.	K_3, K_4
CO 3	Understand the concept and important results related to various algebraic structures	K_2, K_3
CO 4	Understand the concept and important results of Lattice theory	K_3
CO 5	Demonstrate the understanding of Graph theory and recurrence relation and apply in solving the problems.	K_3, K_4
DETAILED SYLLABUS		**3-1-0**

Unit	Topic	Lecture
I	**Set Theory & Relations**: Introduction, Combination of sets. Relations: Definition, Operations on relations, Properties of relations, Composite Relations, Equality of relations, Recursive definition of relation, Order of relations. **Functions**: Definition, Types of functions, Operations on functions. Growth of Functions. **Natural Numbers:** Introduction, Mathematical Induction, variants of Induction, Proof methods: Proof by counterexample and proof by contradiction	**08**
II	**Propositional Logic:** Proposition, well formed formula, Truth tables, Tautology, Satisfiability, Contradiction, Algebra of proposition, Theory of Inference. **Predicate Logic:** First order predicate, well formed formula of predicate, quantifiers, Inference theory of predicate logic	**08**
III	**Algebraic Structures:** Definition, Groups, Subgroups, Order, Cyclic Groups, Cosets, Lagrange's theorem, Normal Subgroups, Permutation and Symmetric groups, Group Homomorphisms, Elementary Properties of Rings and Fields**.**	**08**
IV	**Lattices**: POSET, Hasse Diagram, Properties of lattices – Bounded, Complemented, Distributed, Modular and Complete lattice. **Boolean Algebra:** Axioms and Theorems of Boolean Algebra, Algebraic manipulation of Boolean expressions. Simplification of Boolean Functions, Karnaugh Maps.	**08**
V	**Graphs**: Definition and terminology, Representation of graphs, Multigraphs, Bipartite graphs, Planar graphs, Isomorphism and Homeomorphism of graphs, Euler and Hamiltonian paths, Graph coloring **Combinatorics**: Introduction, Counting Techniques, Pigeonhole Principle, Recurrence Relation: Recursive definition of functions, Recursive algorithms, Method of solving recurrences, generating Functions.	**08**

Text books:

1. Kenneth H. Rosen, Discrete Mathematics and Its Applications, 7/e, McGraw-Hill.
2. Liptschutz, Seymour, " Discrete Mathematics", McGraw Hill.
3. C. L. liu, "Elements Of Discrete Mathematics", McGraw Hill.
4. Y. N. Singh, "Discrete Mathematical Structures", Wiley India.

Reference books:

1. Koshy, Discrete Structures, Elsevier Pub. 2008
2. B. Kolman, R.C. Busby, and S.C. Ross, Discrete Mathematical Structures, 5/e, Prentice Hall, 2004.
3. R.P. Grimaldi, Discrete and Combinatorial Mathematics, 5/e, Addison Wesley, 2004
4. Trembley, J. P. and R. Manohar, "Discrete Mathematical Structure with Application to Computer Science", McGraw Hill.
5. Deo, Narsingh, "Graph Theory With application to Engineering and Computer Science.", PHI.

ICS 351: Data Structures Lab

List of Experiments

Write C programs for:

1. **Realization of Linked List:** Singly Linked Lists, Circular Linked List, Doubly Linked Lists : Insert, Display, Delete, Search, Count, Reverse(SLL), Polynomial , Addition , Comparative study of arrays and linked list

2. **Realization of Stacks:** Array implementation, Linked List implementation, Evaluation of postfix expression and balancing of parenthesis , Conversion of infix notation to postfix notation

3. **Realization of Queue:** Linked List implementation of ordinary queue, Linked List implementation of circular queue, Array implementation of priority queue

4. **Implementing Sorting Techniques:** Bubble Sort, Insertion Sort, Quick sort

5. **Implementing Searching and Hashing Techniques:** Linear search, Binary search, Methods for Hashing: Modulo Division, Digit Extraction, Fold shift, Fold Boundary, Linear Probe for Collision Resolution. Direct and Subtraction hashing

6. **Implementing Trees:** Binary search tree : Create, Recursive traversal: preorder, post order, inorder, Count number of nodes, Heap Sort

7. **Implementing Graphs**: Represent a graph using the Adjacency Matrix, BFS, Finding the minimum spanning tree using Kruskal's or Prim's Algorithm, Shortest Path Algorithm

   Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students*

ICS 352: Digital Logic and Computer Organization Lab

List of Experiments

1. Verification of truth table of basic logic gates.
2. Realization of HALF ADDER, FULL ADDER, 4- Bit Full Adder
3. Realization of Binary-to-Gray, Gray-to-Binary code conversions.
4. Realization of 3x8 line DECODER.
5. Implementing 4x1 and 8x1 MULTIPLEXERS.
6. Verification of the excitation tables of various FLIP-FLOPS.
7. Design of an 8-bit Input/ Output system with four 8-bit Internal Registers.
8. Design of a counter to count three bit number in following order: 5, 3,7,1, 0, 2, 4
9. Design of an 8-bit Ripple Carry Adder.
10. Design of an 8-bit carry look Ahead Adder
11. Design of an 8-bit Booth Multiplier.

Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students*

ICS 353: IT Tools (Linux Shell programming and LATEX)

List of Experiments

1. Installation of Linux operating system.
2. Understanding and practice of various Linux commands.
3. Write a shell script that receives any number of file names as arguments and checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
4. Illustrate by writing script that will print, message “Hello World, in Bold and Blink effect, and in different colors like red, brown etc using echo commands?
5. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
6. Illustrate by writing script using for loop to print the pyramid patterns?
7. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
8. Installing MiKTeX and text editor (freeware Texmaker/Texworks) to create a LaTeX document.
9. Create simple LaTeX documents.
10. Prepare a scientific report includes Title, Introduction, figures with caption, tables with caption, use of equations, references that uses various features of the LATEX. Sample of the document may be given by the instructor.

Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students*

ICS 401: Operating System

Course Outcome (CO)		Bloom's Level
At the end of course , the student will be able to understand		
CO 1	Understand the structure and functions of OS	K_2
CO 2	Understand the concept of Process, Thread, Scheduling and deadlock and able to solve the problems based upon the concept.	K_2, K_3
CO 3	Understand the concept of paging, segmentation, demand paging and virtual memory of memory management techniques and able to solve the problems based upon the concept	K_2, K_3
CO 4	Understand the principles of concurrency and write solutions of classical synchronization problems	K_2, K_3
CO 5	Understand the file system management, disk scheduling algorithms and I/O management.	K_2

DETAILED SYLLABUS

Unit	Topic	Lecture
I	**Introduction:** Operating system and functions, Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocess Systems, Multithreaded Systems, Computer-System Operation: I/O Interrupt, Dual-Mode Operation, I/O, Memory and CPU protection, Operating System Structure- Layered structure, System Components, Operating System services, System calls, Reentrant Kernels, Monolithic and Microkernel Systems, Virtual Machines,	**08**
II	**CPU Scheduling:** Process Concept, Process Control Block (PCB), Process address space, Process identification information, Process States, Process Transition Diagram, Schedulers, Scheduling Concepts, Performance Evaluation Criteria, Threads and their management, Scheduling Algorithms, Multiprocessor Scheduling. **Deadlock:** System model, Deadlock characterization, Deadlock handling, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from deadlock.	**08**
III	**Memory Management:** Basic bare machine, Resident monitor, Logical address, Physical-Address, address binding, Contiguous memory Allocation, fragmentation, Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Protection schemes, Paging, Segmentation, Multilevel Paging, Segmentation with Paging, **Virtual Memory:** Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Cache memory organization, Locality of reference.	**08**
IV	**Concurrent Processes**: Principle of Concurrency, Producer / Consumer Problem, Mutual Exclusion, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Test and Set operation; Classical Problem in Concurrency- Dining Philosopher Problem, Producer-Consumer Problem, Reader-Writer problem, Inter Process Communication models	**08**
V	**File Management:**: File System: File concept, File organization and access mechanism, File allocation: Contiguous, Linked and indexed allocation, Free space management, File directories, and File sharing, File system implementation issues, File system protection and security. **Disk Scheduling:** Disk storage and disk scheduling: FCFS, SSTF, SCAN, C-SCAN **I/O Management**: Introduction to I/O devices management, I/O polling, Interrupts and DMA	**08**

Text books:

1. Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley

Reference Books:

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

ICS 402: Object Oriented Programming using JAVA

Course Outcome (CO)		Level (KL)
At the end of course , the student will be able to understand		
CO 1	Understand different programming structures in a platform independent language.	K_2, K_3
CO 2	Develop the object-oriented programming concepts using Java.	K_2, K_3
CO 3	Implement exception handling, file handling in Java.	K_2, K_3
CO 4	Implement multithreading and apply new java features to build java programs.	K_2, K_3
CO 5	Analyse java programs with Collection Framework	K_2, K_3

DETAILED SYLLABUS

Unit	Topic	Lecture
I	**Introduction:** Why Java, History of Java, JVM, JRE, Java Environment, Java Source File Structure, and Compilation. **Program Structure in Java:** Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments. Data Types, Variables, and Operators: Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Arrays &strings **Control Statements:** Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.	**08**
II	**Object Oriented Programming**: Class, Object, Inheritance Super Class, Sub Class, Overriding, Overloading, Encapsulation, Polymorphism, Abstraction, Interfaces, and Abstract Class. **Packages:** Defining Package, CLASSPATH Setting for Packages, Making JAR Files for Library Packages, Import and Static Import Naming Convention For Packages processor evolution and types, microprocessor architecture and operation of its components, addressing modes, interrupts, data transfer schemes, instruction and data flow, timer and timing diagram, Interfacing devices. **Methods:** Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods. Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.	**08**
III	**Exception Handling**: The Idea behind Exception, Exceptions & Errors, Types of Exception, Control Flow in Exceptions, JVM Reaction to Exceptions, Use of try, catch, finally, throw, throws in Exception Handling, In-built and User Defined Exceptions, Checked and Un-Checked Exceptions. **Input /Output Basics**: Byte Streams and Character Streams, Reading and Writing File in Java.	**08**
IV	**Multithreading:** Thread, Thread Life Cycle, Creating Threads, Thread Priorities, Synchronizing Threads, Inter-thread Communication. **Java New Features:** Functional Interfaces, Lambda Expression, Method References, Stream API, Default Methods, Static Method, Base64 Encode and Decode, ForEach Method, Try-with-resources, Type Annotations, Repeating Annotations, Java Module System, Diamond Syntax with Inner Anonymous Class, Local Variable Type Inference, Switch Expressions, Yield Keyword, Text Blocks, Records, Sealed Classes	**08**
V	**Java Collections Framework:** Collection in Java, Collection Framework in Java, Hierarchy of Collection Framework, Iterator Interface, Collection Interface, List Interface, ArrayList, LinkedList, Vector, Stack, Queue Interface, Set Interface, HashSet, LinkedHashSet, SortedSet Interface, TreeSet, Map Interface, HashMap Class, LinkedHashMap Class, TreeMap Class, Hashtable Class, Sorting, Comparable Interface, Comparator Interface, Properties Class in Java.	**08**

Books

1. Herbert Schildt, "Java The complete reference", McGraw Hill Education
2. Craig Walls, “Spring Boot in Action” Manning Publication
3. Steven Holzner, “Java Black Book”, Dreamtech.
4. Balagurusamy E, “Programming in Java”, McGraw Hill
5. Java: A Beginner’s Guide by Herbert Schildt, Oracle Press

ICS 403: Theory of Automata and Formal Languages

Course Outcome (CO)		Bloom's Level
At the end of course , the student will be able to:		
CO 1	Understand the concepts of abstract machine, formal languages, FA and obtain minimized DFA for any given NFA	K_3, K_4
CO 2	Write regular expressions for the FA and apply pumping lemma for proving a language non-regular	K_3, K_4
CO 3	Write context free grammar and convert them in normal forms.	K_3, K_4
CO 4	Able to design PDA and apply CYK algorithm for testing membership	K_3, K_4
CO 5	Design Turing machine and demonstrate the understanding of recursive and recursively enumerable languages, decidability, P and NP Class complexity	K_3, K_4
DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Lectures**
Unit-I	**Finite Automata:** Introduction: Automata, Computability and Complexity, Alphabet, Symbol, String, Formal Languages, Non Deterministic Finite Automaton (NFA) and Deterministic Finite Automaton (DFA)- Definition, Representation, Language acceptance, Equivalence of DFA and NFA with and without ε-Transition, Minimization of Finite Automata. Finite Automata with output- Moore and Mealy Machine, Equivalence of Moore and Mealy Machine.	**08**
Unit-II	**Regular Expressions and Languages:** Regular Expressions, Transition Graph, Kleen's Theorem, Finite Automata to Regular Expression- Arden's theorem, Algebraic Method Using Arden's Theorem, Regular and Non-Regular Languages- Closure properties of Regular Languages, Pigeonhole Principle, Pumping Lemma for regular languages, Application of Pumping Lemma, Decision properties of Finite Automata and Regular Languages	**08**
Unit-III	**Regular and Context Free Grammar**: Regular Grammars, Regular grammar and Finite Automata, CFG: Definition, Derivations, Languages, Derivation Trees, Ambiguity, Simplification of CFG, Left recursion, Normal Forms of CFG- Chomsky Normal Form(CNF), and Greibach Normal Form (GNF), Chomsky Hierarchy,	**08**
Unit-IV	**Push Down Automata and Context Free Languages**: Nondeterministic Pushdown Automata (NPDA) and Deterministic Pushdown Automata(DPDA): Definition, Moves, Language acceptance by Final state, Language acceptance by Empty stack, Deterministic Context free Languages(DCFL), PDA for CFG, CFG to PDA, Pumping Lemma for CFL, Closure properties of CFL, Two stack Pushdown Automata, Decision Problems of CFL, CYK Algorithm	**08**
Unit-V	**Turing Machines**: Basic Turing Machine Model, Representation of Turing Machines, Language Acceptability of Turing Machines, Techniques for Turing Machine Construction, Variants of Turing Machine, Turing Machine as Computer of Integer Functions, Universal Turing machine, Halting problem, Recursive and Recursively Enumerable language, Properties, Church's Thesis, Halting Problem, Decidability, Post Correspondence Problem, Introduction to Recursive Function Theory. Introduction to P and NP Complexity Classes	**08**

Text books:

1. Introduction to Automata theory, Languages and Computation, J. E. Hopcraft, R. Motwani, and Ullman. 2nd Edition, Pearson Education Asia

Reference Books:

1. Introduction to languages and the theory of computation, J Martin, 3rd Edition, Tata McGraw Hill
2. Introduction to the Theory of Computation, Michael Sipser 3rd Edition, Cengage Learning.
3. Elements and Theory of Computation, C Papadimitrou and C. L. Lewis, 2nd Edition, Pearson Education
4. Theory of Computer Science: Automata, Languages and Computation, K.L.P. Mishra and N. Chandrasekaran, 3rd Edition, PHI

ICS 451: Operating System Lab

List of Experiments

1. Study of hardware and software requirements of different operating systems (LINUX/ WINDOWS)
2. Execute various LINUX system calls for
    i. Process management
    ii. File management
    iii. Input/output Systems calls
3. Write C Programs to demonstrate CPU Scheduling Policies:
    i. SJF
    ii. Priority
    iii. FCFS
    iv. Multi-level Queue
4. Write C Programs to demonstrate various file storage allocation technique:
    i. Contiguous(using array)
    ii. Linked –list(using linked-list)
    iii. Indirect allocation (indexing)
5. Write C Programs to demonstrate various page replacement techniques: FIFO, LRU, Optimal
6. Write C Programs to demonstrate external and internal fragmentation
    i. Free space list of blocks from system
    ii. List process file from the system
7. Write C Programs to demonstrate the compaction for the continually changing memory layout and calculate total movement of data
8. Write C Programs to create resource allocation graph RAG)
9. Write C Programs to implement of Banker's algorithm for avoiding deadlock
10. Write C Programs for conversion of resource allocation graph (RAG) to wait for graph (WFG)
11. Write C Programs to implement to obtain the solution for Bounded Buffer Producer-Consumer problem using semaphores
12. Write C Programs to implement to obtain the solution for Reader-Writer problem using semaphores

Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students.*

ICS 452: Object Oriented Programming using JAVA Lab

List of Experiments

1. Use Java compiler and eclipse platform to write and execute java program.
2. Creating simple java programs using command line arguments
3. Write a java program to create a simple calculator.
4. Understand OOP concepts and basics of Java programming.
5. Create Java programs using inheritance and polymorphism.
6. Implement error-handling techniques using exception handling and multithreading.
7. Create java program with the use of java packages.
8. Construct java program using Java I/O package.
9. Create three threads created from a main method. Each thread should perform different tasks.
10. Write a program to make your main thread wait for termination till the child threads are not finished.

Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students.*

ICS 453: Python Programming Lab

List of Experiments

1. Write a program to demonstrate different number data types in Python.
2. Create a list and perform the following methods a) insert() b) remove() c) append() d) len() e) pop() f) clear()
3. Create a dictionary and apply the following methods a) print the dictionary items b) access items c) use get() d)change values e) use len()
4. Create a tuple and perform the following methods a) Add items b) len() c) check for item in tuple d)Access items
5. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
6. Create a menu driven Python program with the following options 1. ADDITITON 2. SUBTRACTION 3. MULTIPICATION 4. DIVISION. The program should take input from users and perform the operation accordingly. Use functions with arguments.
7. Write a python program to find largest of three numbers.
8. Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9]
9. Write a Python program to construct the following pattern, using a nested for loop

a.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

b.

```
    A
   ABC
  ABCDE
 ABCDEFG
ABCDEFGHI
```

10. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
11. Write a python program to print date, time using date and time functions.
12. Using a numpy module create an array and check the following: a. Type of array b. Axes of array c. Shape of array d. Type of elements in array.
13. Using a numpy module create array and check the following: a. Reshape 3X4 array to 2X2X3 array b. Sequence of integers from 0 to 30 with steps of 5 c. Flatten array d. Constant value array of complex type.
14. Write a python program to concatenate the dataframes with two different objects.
15. Write a python code to read a csv file using pandas module and print the first and last five lines of a file.

Note: *The list is indicative. The instructor may add/ modify the list as per his wisdom for better hands-on of the students.*

ICS 454: Web Technology based Mini Project

Note: The students will be assigned one or two relatively long assignments/ Mini Project utilizing Web Technology to solve the problems related to (but not limited) health, environment, agriculture, smart city, education, and sustainable living etc.

OPEN ELECTIVES
[To be offered by CSE Department]

- **IOE 031 / IOE 041: Basics of Data Structures and Algorithm**
- **IOE 032 / IOE 042: Computer Based Numerical Techniques**

SYLLABI
IOE 031 / IOE 041: Basics of Data Structures and Algorithms

Course Outcome (CO)		Bloom's Level
At the end of course, the student will be able to understand		
CO 1	Understand linear data structures, their implementation and application	**K_3, K_4**
CO 2	Understand stack and queue and demonstrate the knowledge in the application to the problems involving recursion	**K_3, K_4**
CO 3	Implementation of binary tree, tree operations including traversal.	**K_2, K_3**
CO 4	Understanding Graph, graph operations, implementation and applications.	**K_2, K_3**
CO 5	Understand and analyze the complexity of various sorting algorithms	**K_3, K_4**

DETAILED SYLLABUS		
Unit	**Topic**	**Lecture**
I	**Introduction:** Basic Terminology, Elementary Data Organization, Built in Data Types in C, Efficiency of an Algorithm, Asymptotic notations, Abstract Data Types (ADT) **Arrays**: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order. **Linked lists:** Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable & Two variables Polynomial.	08
II	**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Fibonacci numbers, and Hanoi towers, Tradeoffs between iteration and recursion. **Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.	08
III	**Trees:** Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Binary Search Tree, Strictly Binary Tree, Complete Binary Tree, Extended Binary Trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Constructing Binary Tree from given Tree Traversal, Insertion , Deletion, Searching & Modification of data in Binary Search tree.	08
IV	**Graphs:** Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Graph Traversal: Depth First Search and Breadth First Search, Minimum Spanning Trees, Prim's and Kruskal's algorithm. Shortest Path algorithms.	08
V	**Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing. **Sorting:** Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort and Radix Sort.	08

Text books:

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI Learning Private Limited, Delhi India
2. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.

Reference books:

1. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
2. Gilberg, Forouzan, Data Structures: A Pseudocode Approach with C 3rd edition , Cengage Learning publication
3. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education.
4. Berztiss, AT: Data structures, Theory and Practice, Academic Press.
5. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGraw Hill.

IOE 032 / IOE 042: Computer Based Numerical Techniques

Course Outcome (CO)		Bloom's Level
At the end of course , the student will be able to:		
CO 1	Understand the concept of errors in numerical computation and able to apply methods to find roots of equations	K_3, K_4
CO 2	Apply the methods and write computer algorithms to find interpolating polynomials for the given data	K_3, K_4
CO 3	Analyse and apply the methods to find the best fit polynomial and non polynomial curves for the given data	K_3, K_4
CO 4	Apply the methods, analyze the convergence and write algorithms to solve given system of equations	K_3, K_4
CO 5	Remember the concept of formula based the solution of differentiation, integration methods and ordinary differential equations and write computer algorithms	K_3
DETAILED SYLLABUS		
Unit	**Topic**	**Lecture**
I	**Introduction to Computer Arithmetic and Errors:** Floating Point Arithmetic, Machine epsilon, Round off Error, Chopping Error, Truncation Error, Associative and Distributive Law in Floating Point arithmetic, Inherent Error, Error propagation, Numerical Instability **Roots of Equation:** Iterative Methods: Bisection method, Regula-Falsi method, Newton-Raphson method rate of convergence, Aitken Acceleration of Convergence, Modified Newton Raphson Method for Multiple roots, Sturm theorem, Birge-Vieta Method for Polynomials, Computer Algorithms of these methods	**08**
II	**Interpolation:** Newton's forward difference and backward difference Interpolation, formula. Interpolation with unequal intervals: Langrange's Interpolation, Newton Divided difference formula, Algorithms and Error Analysis of Lagrange and Newton divided difference interpolations, Relationship in various difference operators, Piecewise Linear Interpolation, Cubic Spline Interpolation, Natural Spline	**08**
III	**Curve fitting:** Least Squares Approximation, Linear and Non linear Regression, Multiple regression, ill Conditioning in Least Squares Methods, Gram-Schmidt Process of Orthogonalization. Computer Algorithms of Least Square Curve Fitting, Chebshev Polynomial Approximations, Lanczos Economization of Power Series	**08**
IV	**Solution of Simultaneous Linear Algebraic Equations:** Direct methods: Gauss Elimination Method, ill Conditioned Systems, Condition Number, Iteration Methods: Jacobi Method, Gauss-Siedal method, Successive Over Relaxation Method, Rate of Convergence	**08**
V	**Numerical Differentiation:** Introduction, Numerical differentiation based on interpolation and finite differences **Numerical Integration:** Trapezoidal rule, Simpson's 1/3 and 3/8 rule, Newton cotes Integration, Gaussian Integration: Gauss Legendre and Lobatto Methods, Error analysis of Integration Methods, Computer Algorithms **Solution of differential Equations**: Picard's Method, Euler's Method, Taylor's Method, Runge - Kutta Methods, Predictor Corrector Methods, Stability of solution	**08**

Text books:

1. Jain, Iyengar and Jain, "Numerical Methods for Scientific and Engineering Computations", New Age International.

Reference Books

1. Gerald and Wheatley, "Applied Numerical Analyses", Pearson Education.
2. Rajaraman V, "Computer Oriented Numerical Methods", Pearson Education.

COMMON COURSES

- **INC 301/INC 401: Python Programming**
- **INC 302/INC 402: Computer System Security**

SYLLABI

INC 301 / INC 401: Python Programming

Course Outcome (CO)		Bloom's Knowledge Level (KL)
At the end of course , the student will be able to understand		
CO 1	Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.	K_1, K_2
CO 2	Demonstrate proficiency in the handling of strings and functions	K_1, K_2
CO 3	Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.	K_3
CO 4	Identify the commonly used operations involving file systems and regular expressions.	K_1, K_2
CO 5	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python	K_2, K_3

DETAILED SYLLABUS		
Unit	**Topic**	**Lecture**
I	**Introduction to Python:** Python variables, Python basic Operators, Understanding python blocks. Python Data Types, Declaring and using Numeric data types: int, float etc.	**06**
II	**Python Program Flow Control Conditional blocks:** if, else and else if, Simple for loops in python, For loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else. Programming using Python conditional and loop blocks.	**06**
III	**Python Complex data types:** Using string data type and string operations, Defining list and list slicing, Use of Tuple data type. String, List and Dictionary, Manipulations Building blocks of python programs, string manipulation methods, List manipulation. Dictionary manipulation, Programming using string, list and dictionary in-built functions. Python Functions, Organizing python codes using functions.	**06**
IV	**Python File Operations**: Reading files, Writing files in python, Understanding read functions, read(), readline(), readlines(). Understanding write functions, write() and writelines() Manipulating file pointer using seek Programming, using file operations. .	**06**
V	**Python packages:** Simple programs using the built-in functions of packages matplotlib, numpy, pandas etc. GUI Programming: Tkinter introduction, Tkinter and PythonProgramming, Tk Widgets, Tkinter examples. Python programming with IDE.	**06**

Books:

1. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016
2. Lambert, Fundamentals of Python: First Programs with MindTap, 2nd 1st edition , Cengage Learning publication
3. Charles Dierbach, "Introduction to Computer Science using Python", Wiley, 2015
4. Jeeva Jose & P.SojanLal, "Introduction to Computing and Problem Solving with PYTHON", Khanna Publishers, New Delhi, 2016
5. Downey, A. et al., "How to think like a Computer Scientist: Learning with Python", John Wiley, 2015
6. Mark Lutz, "Learning Python", 5th edition, Orelly Publication, 2013, ISBN 978- 1449355739
7. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410
8. Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009
9. David Beazley, Brian Jones., "Python Cookbook", Third Edition, O'relly Publication, 2013.

INC 302 / INC 402: Computer System Security

Course Outcome (CO)		Bloom's Knowledge Level (KL)
At the end of course, the student will be able to		
CO 1	Remember and understand the basic terminologies and concepts of security threats and cybercrimes.	K_1, K_2
CO 2	Remember and understand the approaches used in OS security, concept of buffer overflow and SQL injection attack	K_1, K_2
CO 3	Remember and understand the TCP layer security, Firewalls and security models	K_1, K_2
CO 4	Remember and understand various cryptographic primitives and their purposes	K_1, K_2
CO 5	Understand the various aspects of cyber attacks, cyber security policies and cyber laws	K_1, K_2

DETAILED SYLLABUS

Unit	Topic	Lecture
I	Overview of Computer Security Concepts : Threats, Active and passive Attack, User Identification and Authentication, Message Integrity, Non-repudiation, Availability, Access Control, Database and Cloud Security, Malicious Software, Proxy Servers and Anonymizers, Phishing, Identity Theft, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan-horses and Backdoors, DoS and DDoS Attacks	**06**
II	Overview of Linux/Unix Security and Database Security**:** Unix/Linux Security Architecture, User Account, Superuser, Group, Login/password, Shadow Password File, The Inode, Permissions for Directories, Acess Control: Set UserID and Set GroupID, Changing Permissions, Changing the Root of the Filesystem, Environment Variables, Audit Logs and Intrusion Detection, Buffer Overflow, Scripting , SQL Injection Attacks	**06**
III	TCP Session Hijacking, TCP SYN Flooding Attacks, Domain Name System, Cache Poisoning Attack, DNS Rebinding Attack, SSL Connection Intrusion Detection, Firewalls and Intrusion Prevention Systems Bell–LaPadula Model and Biba Model of security Management Issues: Security Management and Risk Assessment, Human Resources Security, Legal and Ethical Aspects	**06**
IV	Overview of Cryptographic Techniques: Symmetric Encryption, Decryption, Public Key Cipher, Authentication, Message Authentication Codes, Key Exchanges, Steganography, Digital signatures, Public Key Infrastructure, Electronic Payments Systems	**06**
V	Cybercrimes, Cyber stalking, Botnets. Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Privacy Threats, Challenges in Computer Forensics. Introduction to security policies and cyber laws: Need for An Information Security Policy, Introduction to Indian Cyber Law, Objective and Scope of the Digital Personal Data Protection Act 2023, Intellectual Property Issues, Overview of Intellectual Property Related Legislation in India, Patent, Copyright, Trademarks.	**06**

Text books:

1. Dieter Gollman, "Computer Security", 3rd edition, Wiley.
2. William Stallings, Lawrie Brown, "Computer Security: Principles and Practice", 3rd edition, Prentice Hall.
3. Sunit Belapure and Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics And Legal Perspectives", Wiley India.

Reference books:

1. T. J. Mowbray, "Cybersecurity: Managing Systems, Conducting Testing, and Investigating Intrusions", John Wiley & Sons.
2. J. Graham, R. Olson and R. Howard, "Cyber Security Essentials", CRC Press.
3. Anti Hacker Tool Kit (Indian Edition) by Mike Shema, McGraw-Hill.