Semester 4 Curriculum and Syllabus

SEMESTER-4							
Sl. No.	**Type**	**Course No.**	**Course Name**	**L**	**T**	**P**	**Credits**
THEORY							
1	PC	YCS4001	Computer Organization and Architecture	3	0	0	3
2	PC	YCS4002	Design and Analysis of Algorithms	3	1	0	4
3	PC	YCS4003	Data Base Management System	3	0	0	3
4	PC	YCS4004	Formal Language and Automata	3	0	0	3
5	HS	YMG4001	Economics for Engineers	2	0	0	2
PRACTICAL							
6	PC	YCS4101	Computer Organization and Architecture Laboratory	0	0	3	1.5
7	PC	YCS4102	Algorithms Laboratory	0	0	3	1.5
8	PC	YCS4103	Data Base Management System Laboratory	0	0	3	1.5
9	PC	YCS4104	Programming Practices II	0	0	3	1.5
MANDATORY NON-CGPA COURSE							
10	MC	YCS4501	Constitution of India	3	0	0	0
SESSIONAL (ONLY INTERNAL EVALUATION)							
11	PROJ	YCS4201	Innovative Project II	0	0	3	1.5
TOTAL				**17**	**1**	**15**	**22.5**

Course Code	YCS4001			
Course Title	Computer Organization and Architecture			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Digital Circuits and Logic Design			

Learning Objective:

In this course, the students will learn about the evolution of computer systems and development in computer organization and architecture, and the various functional units of a computer system with special emphasis on how instructions get executed. This course will cover the processor unit, the arithmetic and logic unit, the memory unit and input/output organization.
After the completion of this course, the student will better understand how exactly the programs are executed in a computer system.

Course Outcome:

CO1: To explain the process of instruction execution

CO2: To analyze and design control unit of a computer system

CO3: To analyze and design adder, multiplier and division unit

CO4: To analyze and design memory subsystems

CO5: To explain and classify various input/output data transfer techniques

Course Content:

Module 1: Evolution of Computer System **[4L]**

Introduction to computing system: computer organization and architecture, basic functional units of a computer, evolution of computers, stored-program concept, Von-Neumann and Harvard models

Module 2: Basic Operation of Computer **[7L]**

Instruction Set Architecture: CPU registers, instruction format and encoding, addressing modes, instruction set, instruction types, instruction decoding and execution, basic instruction cycle, Reduced Instruction Set Computer (RISC), Complex Instruction Set Computer (CISC).
Case study: MIPS Instruction set, MIPS assembly language programming.

Module 3: Processor Unit Design **[7L]**

Register transfer operations, internal single and multi-bus architecture.
Design of control unit: hardwired control unit design, microprogrammed control unit design, concept of control word and control store.
Horizontal, vertical and diagonal microprogrammed control unit design.

Module 4: Arithmetic Unit Design **[6L]**

Adder and subtractor, shift-and-add multiplication.
Signed multiplication: Booths algorithm, integer division, restoring and non-restoring division.
Floating point representation: IEEE floating point format, floating point arithmetic.

Module 5: Memory Unit Design **[7L]**

Basic memory types: Random Access Memory (RAM), Read Only Memory (ROM), Static RAM, Dynamic RAM.
Memory hierarchy, Cache memory: mapping techniques, Memory interleaving.

Module 6: Input Output Organization **[5L]**

I/O mapped I/O and Memory mapped I/O, Synchronous and Asynchronous serial data communication. Secondary memory: disk, flash memory.
I/O Data transfer techniques: Programmed I/O, Interrupt-driven I/O, Direct Memory Access (DMA).

Text/Reference Books:

1. C. Hamacher, Z. Vranesic and S. Zaky, "Computer Organization (5th Ed.)", Tata-McGraw-Hill.
2. W. Stallings, "Computer Organization and Architecture (6th Ed.)", Prentice Hall of India.
3. D. A. Patterson, and J. L. Hennessy, "Computer Organization and Design – The Hardware/ Software Interface", Morgan Kaufmann.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	2	-	-	2	-	-	1	-	1	2
CO2	1	1	3	-	-	2	-	-	2	-	1	2
CO3	2	2	2	2	2	1	-	-	1	-	-	2
CO4	2	2	1	2	2	1	-	-	2	-	-	2
CO5	2	1	1	-	1	1	-	-	1	-	-	2

Course Code	YCS4002			
Course Title	Design and Analysis of Algorithms			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	3	1	0	3
Total Contact Hours	36			
Pre-requisites	a)Fundamentals of Programming b) Data Structures and Algorithms			

Learning Objective:

It will covers topics such as algorithm complexity concepts and diverse algorithmic designs such as dividing and conquering, dynamic programming and greedy algorithms. The course will also include important search and sorting algorithms, graphs, and basic approaches of optimization.

Course Outcome:

CO1: To understand the concepts of time and space complexity, worst case, average case and best case complexities and the big-O notation

CO2: To apply design principles and concepts to algorithm design

CO3: To understand and analyze the mathematical foundation in analysis of algorithms

CO4: To explain and classify different algorithmic design strategies

CO5: To analyze the efficiency of algorithms using time and space complexity theory

Course Content:

Module 1: Complexity Analysis **[7L]**

Time and space Complexity, Different asymptotic notations – their mathematical significance. Solving recurrences: substitution method, recurrence tree method, Master Theorem.

Module 2: Divide and Conquer **[9L]**

Basic concept, Examples: binary search, merge sort, quick sort and their complexity (all three cases). Heap sort and its complexity, Karatsuba algorithm.

Lower Bound Theory: Comparisons trees, Oracle and adversary argument, State space method.

Module 3: Dynamic Programming **[14L]**

Basic concepts, matrix chain manipulation, Strassen's algorithm, longest common subsequence, all-pair shortest paths (Floyd Warshall), single-source shortest path (Dijkstra, Bellman-Ford), 0/1 Knapsack problem, Travelling Salesman problem.

Greedy Method: Basic concept, Examples: fractional Knapsack problem, job sequencing with deadlines, minimum cost spanning tree using Prim's and Kruskal's method, Huffman encoding and decoding.

Backtracking: Basic concept, Examples: n-queens problem, graph coloring problem.

Disjoint Set Manipulation: Set manipulation algorithm like UNION-FIND, union by rank.

Module 4: String Matching Problem **[10L]**

Different techniques: Naive algorithm, string matching using finite automata, and Knuth, Morris, Pratt (KMP) algorithm with their complexities.
Amortized Analysis and Network Flow: Aggregate, Accounting, and Potential Method, Ford Fulkerson algorithm, Max-Flow Min-Cut.

Module 5: Notion of NP-Completeness **[8L]**

P class, NP class, NP hard class, NP complete class their interrelationship, Reductions and Polynomial time Reducibility, Satisfiability problem (3-SAT and 2-SAT), Cook-Levin's theorem, Clique decision problem, Vertex Cover problem.

Text/Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms", MIT Press.
2. E. Horowitz and S. Shani, "Fundamentals of Computer Algorithms", Universities Press.
3. K. Mehlhorn and P. Sanders, "Data Structures and Algorithms", Springer.
4. A. Aho, J. Hopcroft and J. Ullman "Design and Analysis of Computer Algorithms", Addison-Wesley.
5. 5. D. E.Knuth, "The Art of Computer Programming (Vol. 3)", Addison-Wesley.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	1	2	-	-	2	-	-	3
CO2	3	3	3	3	1	2	-	-	1	-	2	3
CO3	3	2	2	3	1	2	-	-	2	-	-	3
CO4	3	3	3	3	1	2	-	-	1	-	-	3
CO5	3	2	2	3	1	2	-	-	2	-	-	3

Course Code	YCS4003			
Course Title	Data Base Management System			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Data Structures and Algorithms			

Learning Objective:

In this course, the students will be able to learn the data models, conceptualize and depict a database system; design system using E-R diagram; learn SQL & relational database design; understand the internal storage structures using different file and indexing techniques; know the concepts of transaction processing, concurrency control techniques and recovery procedure.

Course Outcome:

CO1: To apply the knowledge of E-R diagram for an application

CO2: To explain the creation of the normalized relational database model

CO3: To analyze real world queries to generate reports from it

CO4: To determine whether the transaction satisfies the ACID properties

CO5: To create and maintain the database of an organization

Course Content:

Module 1: Introduction **[3L]**

Concept and overview of DBMS, data models.
Database languages, database administrator, database users, three-schema architecture of DBMS.

Module 2: Entity-Relationship and Relational Database Model **[9L]**

Basic concepts, design issues, mapping constraints, keys, entity-relationship diagram, weak entity sets, extended E-R features, case study on E-R model.
Structure of relational databases, relational algebra, relational calculus, extended relational algebra operations, views, modifications of the database.

Module 3: SQL and Integrity Constraints **[6L]**

Concept of DDL, DML, DCL.
Basic structure, set operations, aggregate functions, null values, domain constraints, referential integrity constraints, assertions, views, nested sub-queries.
Database security application development using SQL, stored procedures and triggers.

Module 4: Relational Database Design **[6L]**

AFunctional dependency, Different anomalies in designing a Database. Normalization using functional dependencies, decomposition, Boyce-Codd Normal Form, 3NF.
normalization using multi-valued dependencies, 4NF, 5NF, Case Study.

Module 5: Internals of RDBMS **[6L]**

Physical data structures, query optimization: join algorithm, statistics and cost based optimization. Transaction processing, concurrency control and recovery management: transaction model properties, state serializability, lock base protocols; two phase locking, deadlock handling.

Module 6: File Organization & Index Structures **[6L]**

File and record Concept, placing file records on disk, fixed and variable sized records, Ttypes of single-level index (primary, secondary, clustering).
Multilevel indices, dynamic multilevel indices using B-tree and B+ tree.

Text/Reference Books:

1. R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems", Addison Wesley Publishing.
2. C.J. Date, "Introduction to Database Management", Vol. I, II, III, Addison Wesley.
3. J.D. Ullman, "Principles of Database Systems", Galgottia Publication.
4. G. Jim and R. Address, "Transaction Processing : Concepts and Techniques", Morgan Kauffman.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	2	-	1	-	-	2	-	2	1
CO2	3	3	3	1	-	2	-	-	2	-	1	2
CO3	3	3	3	1	-	1	-	-	2	-	2	1
CO4	3	3	3	1	2	2	-	-	2	-	1	2
CO5	3	2	2	2	-	1	-	-	2	-	2	1

Course Code	YCS4004			
Course Title	Formal Language and Automata Theory			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Discrete Mathematics b) Programming and Data Structure			

Learning Objective:

In this course the students will learn the theory of computation, different formal language classes and their relationships, various techniques to prove or disprove theorems in automata theory using its properties, approaches to determine the decidability and intractability of computational problems. At the end of the course student will be able analyze complex problems and automaton to find solutions of such problems.

Course Outcome:

CO1: To explain the basic properties of formal languages and grammars

CO2: To understand the tools for recognizing different formal languages

CO3: To differentiate between regular, context-free and recursively enumerable languages

CO4: To apply the theory of computation and computational models including decidability and intractability

Course Content:

Module 1: Introduction to Finite Automata **[10L]**

Finite Automata, Alphabets, Strings, Languages, Regular Languages, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation, State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata, Myhill-Nerode Theorem, FA with output - Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA.

Module 2: Properties of Regular Expression **[7L]**

Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non Regular Languages, Pumping Lemma for regular Languages, Application of Pumping Lemma, Closure and decision properties of Regular Languages.

Module 3: Language & Grammar Formalism **[9L]**

Grammars, Regular grammars-Right linear and left linear grammars, Equivalence between regular linear grammar and FA, Context Free Grammar, Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammar, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs - CNF and GNF, Closure properties of CFLs, Decision Properties of CFLs-Emptiness, Finiteness and Membership, Pumping lemma for CFLs.

Module 4: Push Down Automata **[4L]**

PDA Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of PDA and CFG, CFG to PDA, PDA to CFG, Two stack PDA.

Module 5: Turing Machines and Decidability **[6L]**

Basic model, Definition and representation, Instantaneous Description, Language acceptance by TM, Computable functions, Types of Turing machines, Universal TM, Church's Thesis, Recursive and recursively enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs, Post correspondence problem (PCP), Modified PCP.

Text/Reference Books:

1. J. D. Ullman, J. Hopcroft and R. Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 2007
2. P. Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Learning, 2012
3. K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science : Automata, Languages and Computation", Prentice Hall India, 2008
4. M. Sipser, "Introduction to Theory of Computation", Thomson Course Technology, 2006
5. J. C. Martin , "Introduction to Languages and Theory of Computations", McGraw Hill, 2011
6. E. A. Rich, Automata, "Computability and Complexity", Pearson Education, Inc., 2019
7. D. Kozen, "Automata and Computability", Spinger, 1997
8. H. R. Lewis and C. H. Papadimitriou, "Elements of the Theory of Computation", Prentice Hall of India Private Ltd.,1998
9. Z. Kohavi and N. K. Jha, "Switching and Finite Automata Theory", Cambridge University Press, 2010
10. D. I. A. Cohen, "Introduction to computer theory", John Wiley & Sons, Inc., 1986

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	1	-	2	-	-	1	-	-	1
CO2	3	3	1	1	-	1	-	-	2	-	-	1
CO3	3	2	1	1	1	2	-	-	1	-	-	1
CO4	3	2	1	1	1	1	-	-	2	-	-	1

Course Code	YMG4001			
Course Title	Economics for Engineers			
Category	Humanities			
LTP & Credits	L	T	P	Credits
	2	0	0	2
Total Contact Hours	24			
Pre-requisites	None			

Learning Objective:

In this course the students will learn about the managerial economics, basics of accounting and financial management. At the end of the course, the students will be able to make different managerial decisions in terms of economics and also able to solve financial statement as well as they can make different financing decision for business and at personal level.

Course Outcome:

CO1: To apply the appropriate engineering economics analysis method(s) for problem solving: present worth, annual cost, rate-of-return, payback, break-even, benefit-cost ratio

CO2: To evaluate the cost effectiveness of individual engineering projects using the methods learned and draw inferences for the investment decisions

CO3: To compare the life cycle cost of multiple projects using the methods learned, and make a quantitative decision between alternate facilities and/or systems

CO4: To evaluate the profit of a firm, carry out the break-even analysis and employ this tool to make production decision

CO5: To discuss and solve advanced economic engineering analysis problems including taxation and inflation

Course Content:

Module 1: Introduction **[3L]**

Managerial Economics, Relationship with other disciplines, Firms: Types, Objectives and goals, Managerial Decisions-Decision Analysis.

Module 2: Demand and Supply Analysis **[5L]**

Demand: Types of demand, determinants of demand, Demand function, Demand Elasticity, Demand forecasting. Supply: Determinants of supply, Supply function, Supply Elasticity.

Module 3: Cost Analysis **[5L]**

Element of costs, Marginal cost, Marginal Revenue, Sunk cost, Opportunity cost, Break-even analysis – PV ratio.

Module 4: Elementary Economic Analysis **[4L]**

Inflation: Meaning of inflation, types, causes, measures to control inflation.
National Income: Definition, Concepts of national income, Method of measuring national income.

Module 5: Financial Accounting **[5L]**

Concepts and Definition of Accounting, Journal, Ledger, Trial Balance.
Trading A/C, Profit & Loss A/C and Balance Sheet.

Module 6: Investment Decision **[2L]**

Time value of money: Interest - Simple and compound, nominal and effective rate of interest, Cash flow diagrams, Principles of economic equivalence. Evaluation of engineering projects-Present worth method, Future worth method, Annual worth method, Internal rate of return method, Cost benefit analysis for public projects.

Text/Reference Books:

1. B. Riggs and S.U. Randhwa, "Engineering Economics", McGraw Hill Education India.
2. D. Vengedasalam and K. Madhavan, "Principles of Economics", Oxford University Press.
3. W. G. Sullivan, E. M. Wicks and C. P. Koelling, "Engineering Economy", Pearson.
4. R. P. Seelvan, "Engineering Economics", Prentice-Hall of India.
5. H. L. Ahuja, "Principles of Micro Economics", S. Chand & Company Ltd.
6. S. P. Gupta, "Macro Economics", Tata McGraw Hill.
7. K. K. Dewett, "Modern Economic Theory", S. Chand & Company Ltd.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	1	-	-	-	2	-	-	-	-	2	1
CO2	-	-	-	3	-	2	-	-	-	-	-	1
CO3	-	1	-	-	-	2	-	-	-	-	3	1
CO4	-	-	-	-	-	2	-	-	3	-	-	1
CO5	-	1	-	-	-	2	-	-	-	-	1	1

Course Code	YCS4101			
Course Title	Computer Organization and Architecture Laboratory			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a)Digital Circuits Laboratory			

Learning Objective:

In this laboratory course, the students will be conducting experiments using a MIPS instruction set simulator. They will also learn how to model various hardware blocks using the hardware description language Verilog. They shall be designing various functional units like adder, multiplier, processor, etc. using a Verilog.

Course Outcome:

CO1: To understand how to write assembly language programs in MIPS

CO2: To design various combinational and sequential circuits using Verilog

CO3: To design and analyze various CPU functional units using Verilog

CO4: To apply a pipelined processor using Verilog

Course Content:

1. Familiarization with MIPS assembly language programming using some instruction set simulator like QtSPIM.
   a. Reading and displaying an arbitrary string, and an integer.
   b. Store numbers sequentially in memory and find the minimum, maximum, and sum.
   c. Sort a set of numbers stored in memory. **[2 days]**

2. Familiarization of function calls with MIPS assembly language programming.
   a. Write a function to compute the factorial of a given number.
   b. Write a function to compute the GCD of two numbers.
   c. Write a function to compute the N-th Fibonacci number. **[2 days]**

3. Familiarization with a Verilog simulator like iVerilog, and write simple combinational and sequential modules using behavioral and structural modeling with Verilog.
   a. Write a module to implement an arbitrary Boolean function (e.g. F = A'BC + C'D).
   b. Write a module to implement a full adder, and hence a 4-bit ripple carry adder.
   c. Write a module to implement a D flip-flop, and hence a 4-bit shift register.
   d. Write a module to implement an 8-bit up-down counter with asynchronous clear. **[2 days]**

4. Write Verilog modules to implement functional blocks used in computer organization.
   a. Write a module to implement a 16-bit arithmetic and logic unit with 8 functions.
   b. Write a module to implement read/write operations in a 1024 x 16 memory system. **[2 days]**

5. Implement the MIPS 5-stage pipeline in Verilog, using a subset of 16 instructions. The design has to be tested by writing a test bench containing sample machine language programs stored in a memory module. **[4 days]**

Text/Reference Books:

1. qtSPIM simulator, http://spimsimulator.sourceforge.net/
2. MIPS overview, https://tams.informatik.uni-hamburg.de/applets/hades/webdemos/mips.html
3. M. M. Mano and M. D. Ciletti, "Digital Design: with an Introduction to Verilog HDL (5th Ed.)", Pearson Education.
4. J. Bhasker, "Verilog HDL Synthesis: A Practical Primer", B. S. Publications.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	2	-	-	-	-	-	2	-	-	2
CO2	-	1	2	2	1	-	-	-	2	-	-	2
CO3	1	1	1	2	1	1	-	-	2	-	-	2
CO4	-	-	1	1	2	2	-	-	2	-	-	2

Course Code	YCS4102			
Course Title	Algorithms Laboratory			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a) Programming Practices I			

Learning Objective:

The course aims to provide strategies (divide and conquer, dynamic, greedy) to solve problems in computer effectively. Using the many paradigms of solving problems, the innovative and effective approaches of solving a specific situation will be demonstrated. In each case, the focus is on the rigorous proof of the algorithm's validity.

Course Outcome:

CO1: To prove the correctness and analyze the running time of the basic algorithms

CO2: To design algorithms using the dynamic programming, greedy method, Backtracking, Branch and Bound strategy, and recite algorithms that employ this strategy

CO3: To compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem

CO4: To Identify and analyze criteria and specifications appropriate to new problems

Course Content:

1. Experiments on Divide and Conquer Approach.
   Binary Search (Recursive & Iterative).
   Merge Sort, Heap Sort, Quick Sort.
   Find Maximum and Minimum element from an array of integers. **[2 days]**

2. Experiments on Dynamic Programming.
   Minimum number of scalar multiplications needed for chain of matrix.
   All pair of shortest paths for a graph.
   Single-source shortest path for a graph (Dijkstra, Bellman Ford).
   Longest common subsequence problem. **[2 days]**

3. Experiments on Backtracking.
   The n-Queens problem.
   Graph Coloring problem. **[2 days]**

4. Experiments on Greedy Methods.
   Knapsack problem.
   Job sequencing with deadlines.
   Minimum cost spanning tree by Prim's and Kruskal's algorithm. **[2 days]**

5. Innovative Experiments
   Take the university time table for all departments. Write a computer program to find all conflicts within the time table using graph colouring approach. Provide a solution using Backtracking. Compute the distance and find the stoppages every classmate of yours cover to

reach the institute. Then assume their speeds based on their travelling modes. Compute each student's minimum time to reach the institute premises.

[2 days]

Text/Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms", MIT Press.
2. E. Horowitz and S. Shani, "Fundamentals of Computer Algorithms", Universities Press.
3. K. Mehlhorn and P. Sanders, "Data Structures and Algorithms", Springer.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	3	1	1	-	-	2	-	-	3
CO2	3	2	2	3	1	-	-	-	2	-	-	3
CO3	3	3	2	3	1	1	-	-	2	-	-	3
CO4	3	3	2	1	1	-	-	-	2	-	-	3

Course Code	YCS4103			
Course Title	Data Base Management System Laboratory			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a)Digital Circuits Laboratory			

Learning Objective:

In this course, the students will able to learn the data models, conceptualize and depict a database system; learn the fundamental concepts of SQL queries; understand the concept of designing a database with the necessary attributes; know the methodology of Accessing, Modifying and Updating data & information from the relational databases; learn database design as well as to design user interface and how to connect with database.

Course Outcome:

CO1: To understand the basic concepts regarding database, SQL queries

CO2: To explain the concepts of PL/SQL

CO3: To differentiate between DBMS and advanced DBMS

CO4: To analyze database system concepts and apply normalization to the database

CO5: To apply and create different transaction processing and concurrency control applications

Course Content:

1. Experiments on fundamentals of database systems
   Creating a Database
   Creating a Table
   Specifying Relational Data Types
   Specifying Constraints
   Creating Indexes **[2 days]**

2. Experiments on database Tables and Record handling
   INSERT statement
   Use of SELECT and INSERT together
   DELETE, UPDATE, TRUNCATE statements
   DROP, ALTER statements **[2 days]**

3. Experiments on retrieving data from database
   The SELECT statement
   Use of the WHERE clause
   Use of the Logical Operators in the WHERE clause
   Use of IN, BETWEEN, LIKE , ORDER BY, GROUP BY and HAVING Clause
   Use of the Aggregate Functions
   Combining tables using JOINS
   Sub-queries **[3 days]**

4. Experiments on Miscellaneous Database Management
   Creating Views
   Creating Column Aliases
   Creating Database Users
   Use of GRANT and REVOKE **[1 day]**

5. Experiments on PL/SQL
   Use of decision making statement, different loop structures to solve simple programs (e.g., sum of few numbers, pattern prints, etc.).
   Inserting values into tables, reading data from a table.
   Basic working with CURSORS **[1 day]**

6. Innovative Experiments
   Case study of handling complex databases (e.g., College Management System, Hospital management System, Library management System, Payroll management System, etc.) **[3 days]**

Text/Reference Books:

1. H. F. Korth and A. Silberschatz, "Database System Concepts", McGraw Hill.
2. E. Ramez and S. Navathe, "Fundamentals of Database Systems", Benjamin Cummings Publishing Company.
3. C. J. Date, "Introduction to Database Management", Vol. I, II, III, Addison Wesley.
4. G. Jim and R. Address, "Transaction Processing : Concepts and Techniques", Moragan Kauffman.
5. J.D. Ullman, "Principles of Database Systems", Galgottia Publication.
6. I. Bayross , "SQL, PL/SQL the Programming Language of Oracle", BPB Publications.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	1	-	-	-	-	1	-	-	1
CO2	3	2	2	1	2	-	-	-	1	-	-	1
CO3	1	2	3	-	-	-	-	-	1	-	-	2
CO4	3	1	2	2	1	-	-	-	1	-	1	2
CO5	2	2	3	1	-	-	-	-	1	-	1	2

Course Code	YCS4104			
Course Title	Programming Practices II			
Category	Professional Core			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a) Fundamentals of Programming b) Basic Problem Solving			

Learning Objective:

In this practical course, the students will be learning Python programming basics and paradigm. python looping, control statements and string manipulations. Students will be made familiar with the concepts of various modules, packages and python libraries used for various applications (Machine learning, Deep learning etc.).

Course Outcome:

CO1: Understand and explain the basic principles of Python programming language and object oriented concept.

CO2: Define and demonstrate the use of built-in data structures along with the help of condition checking and looping structures.

CO3: Understand and apply various applications of different modules and packages in Python.

CO4: Learn to handle exceptions and files in Python.

Course Content:

1. History, Features, Setting up path, working with Python, Basic Syntax, Variable and Data Types, Operator. **[1 day]**
2. Conditional Statements: If, If- else, Nested if-else, Looping, For, While, Nested loops , Control Statements : Break, Continue, Pass. **[1 day]**
3. String Manipulation: Accessing Strings, Basic Operations, String slices, Function and Methods. Lists: Introduction, Accessing list, Operations, Working with lists, Function and Methods. **[2 days]**
4. Tuple: Introduction, Accessing tuples, Operations, Working, Functions and Methods. Dictionaries: Introduction, Accessing values in dictionaries, Working with dictionaries, Properties. **[2 days]**
5. Functions: Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables. **[1 day]**
6. Modules: Importing module, Math module, Random module, Packages, Composition, Input-Output Printing on screen, Reading data from keyboard, Opening and closing file, Reading and writing files, Functions. **[2 days]**
7. Exception and File Handling: Exception, Exception Handling, Except clause, Try & finally clause, User Defined Exceptions. **[1 day]**

8. A case study on using a computer game for teaching data structures on stacks and queues. The computer game is developed to help students visualize the data structures and data access operations on stacks and queues. This game-based learning is engaging, fun and, more importantly, abstract concepts in data structures can be visualized and learnt through game playing. **[2 days]**

Text/Reference Books:

1. T. R. Padmanabhan, "Programming with Python (1st Ed.)", Springer.
2. R. Thareja, "Python Programming: using Problem Solving Approach (1st Ed.)", Oxford University Press.
3. W. McKinney, "Python Data Analysis (2nd Ed.)", O.Reilly.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	-	-	-	-	2	1	-	1	-	-	-
CO2	3	1	1	-	1	2	1	-	-	-	-	2
CO3	3	3	1	1	1	2	1	-	1	-	-	2
CO4	3	2	2	1	1	2	1	-	-	-	-	2

Course Code	YCS4501			
Course Title	Constitution of India			
Category	Mandatory Non-CGPA Course			
LTP & Credits	L	T	P	Credits
	3	0	0	0
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

Upon completion of this lesson, students will be able to understand the emergence and evolution of Indian Constitution. Understand and analyse federalism in the Indian context. Understand and analyse the three organs of the state in the contemporary scenario. Understand and Evaluate the Indian Political scenario amidst the emerging challenges.

Course Outcome:

CO1: Develop human values , create awareness about law ratification and significance of Constitution

CO2: Comprehend the Fundamental Rights and Fundamental Duties of the Indian Citizen to implant morality, social values and their social responsibilities.

CO3: Create understanding of their Surroundings, Society, Social problems and their suitable solutions

CO4: Demonstrate with distribution of powers and functions of Local Self Government.

CO5: Realize the National Emergency, Financial Emergency and their impact on Economy of the country.

Course Content:

1. Meaning of the constitution law and constitutionalism **[3L]**
2. Historical perspective of the Constitution of India **[2L]**
3. Salient features and characteristics of the Constitution of India **[1L]**
4. Scheme of the fundamental rights **[2L]**
5. The scheme of the Fundamental Duties and its legal status **[2L]**
6. The Directive Principles of State Policy – Its importance and implementation **[2L]**
7. Federal structure and distribution of legislative and financial powers between the Union and the States **[3L]**
8. Parliamentary Form of Government in India – The constitution powers and status of the President of India **[2L]**
9. Amendment of the Constitutional Powers and Procedure **[2L]**
10. The historical perspectives of the constitutional amendments in India **[2L]**
11. Emergency Provisions: National Emergency, President Rule, Financial Emergency **[3L]**

12. Local Self Government – Constitutional Scheme in India **[3L]**
13. Scheme of the Fundamental Right to Equality **[3L]**
14. Scheme of the Fundamental Right to certain Freedom under Article 19 **[3L]**
15. Scope of the Right to Life and Personal Liberty under Article 21. **[3L]**

Text/Reference Books:

1. D.D. Basu, V.R. Manohar, B.P.Banerjee, S.A.Khan, , Introduction to the Constitution of India. Wadhwa, 2001.
2. P. M. Bakshi & S. C. Kashyap, he constitution of India. Universal Law Publishing, 1982.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	1	-	-	3	2	3	-	-	-	2
CO2	-	-	1	-	-	3	2	3	-	-	-	2
CO3	-	-	1	-	-	3	2	3	-	1	-	2
CO4	-	-	1	-	-	3	2	3	-	1	-	2
CO5	-	-	1	-	-	3	2	3	-	1	-	2