Syllabi

Of

B.Tech in Computer Science and Engineering

Department of Computer Science and Engineering

National Institute of Technology Durgapur

FIRST and SECOND SEMESTER

Semester - I							
Sl. No	Code	Subject	L	T	S	C	H
1	MAC01	Mathematics – I	3	1	0	4.0	4
2	PHC01	Engineering Physics	2	1	0	3.0	3
3	CYC01	Engineering Chemistry	2	1	0	3.0	3
4	XEC01	Engineering Mechanics	2	1	0	3.0	3
5	ESC01	Environmental Science	2	0	0	2.0	2
6	XES51	Engineering Graphics	1	0	3	2.5	4
6	HSS51	Professional Communication Laboratory	1	0	2	2.0	3
7	PHS51	Physics Laboratory	0	0	2	1.0	2
8	CYS51	Chemistry Laboratory	0	0	2	1.0	2
9	WSS51	Workshop Practice	0	0	3	1.5	3
10	XXS51	Co-curricular Activities - I	0	0	2	1.0	2
		TOTAL	13	4	14	24.0	31
Semester - II							
Sl. No	Code	Subject	L	T	S	C	H
1	MAC02	Mathematics – II	3	1	0	4.0	4
2	CSC01	Introduction to Computing	2	1	0	3.0	3
3	ECC01	Basic Electronics	2	1	0	3.0	3
4	EEC01	Electrical Technology	2	1	0	3.0	3
5	BTC01	Life Science	2	0	0	2.0	2
6	XES52	Graphical Analysis using CAD	0	0	2	1.0	2
7	CSS51	Computing Laboratory	0	0	2	1.0	2
8	ECS51	Basic Electronics Laboratory	0	0	2	1.0	2
9	EES51	Electrical Technology Laboratory	0	0	2	1.0	2
10	XXS52	Co-curricular Activities – II	0	0	2	1.0	2
		TOTAL	11	4	10	20.0	25

CSC 01 Introduction to Computing 2-1-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC01</td><td>Introduction to Computing</td><td>PCR</td><td>2</td><td>1</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">None</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: Introduction of basics of computers, languages and different types of software (e.g.,operating system, compilers)
• CO2: Illustrate the flowchart and describe an algorithm for a given problem.
• CO3: Description of different types of operators used in C programming
• CO4: Develop conditional and iterative statements to write C programs.
• CO5: Exercise user defined functions to solve various types of problems
• CO6: Developing C programs that use Pointers to access arrays, strings and functions.
• CO7: Exercise user defined data types including structures and unions to solve problems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Fundamentals of Computer: History of Computer, Generation of Computer, Classification of Computers Basic Anatomy of Computer System, Primary & Secondary Memory, Processing Unit, Input & Output devices. (2L)
Languages: Assembly language, high level language, compiler and assembler (basic concepts). (1L)
Binary and allied number systems representation of signed and unsigned numbers. BCD, ASII. Binary Arithmetic & logic gates. (2L)
Basic concepts of operating systems like MS DOS, MS WINDOW, UNIX, Algorithm & flowchart. (2L)
C Fundamentals: The C character set identifiers and keywords, data type & sizes, variable names, declaration, statements. (2L)
Operators & Expressions: Arithmetic operators, relational and logical operators, type, conversion, increment and decrement operators, bit wise operators, assignment operators and expressions, precedence and order of evaluation. Input and Output: Standard input and output, formatted output -- printf, formatted input scanf. (8L)
Flow of Control: Statement and blocks, if - else, switch, loops - while, for, do while, break and continue, go to and labels. (5L)
Fundamentals and Program Structures: Basic of functions, function types, functions returning values, functions not returning values, auto, external, static and register Variables, scope rules, recursion, function prototypes, C pre-processor, command line arguments. (5L)
Arrays and Pointers: One dimensional, two dimensional arrays, pointers and functions, multi-dimensional arrays. (10L)
Structures Union and File: Structure, union, structures and functions, arrays of structures, file read, file write. (5L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. E Balagurusamy, "Computing Fundamentals and C Programming", McGraw Hill Education; Second edition (2017).
2. Y. Kanetkar, "Let Us C", BPB Publications, Sixteenth edition (2017).</td></tr>
</table>

	3. E. Balagurusamy, "Programming in ANSI C", McGraw Hill Education India Private Limited, Seventh edition (2017). **Reference Books:** 1. B. S. Gottfried, "Programming with C", McGraw Hill Education, Fourth edition (2018). 2. B. W. Kernighan, D. Ritchie, "The C Programming Language", Pearson Education India, Second edition (2015).

CSS 51 Computing Laboratory 0-0-2 1Credit 2Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS51</td><td>Computing Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>2</td><td>2</td><td>1</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">None</td><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: To understand the principle of operators, loops and branching statements.
• CO2: Implementation of function, recursion, arrays, and pointers based several types of assignments.
• CO3: To detail out the operations of strings.
• CO4: To understand structure and union.
• CO5: Application of C-programming to solve various types of problems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">List of Experiments:

1. Assignments on expression evaluation.
2. Assignments on conditional branching, iterations, pattern matching.
3. Assignments on function, recursion.
4. Assignments on arrays, pointers, parameter passing.
5. Assignments on string using array and pointers.
6. Assignments on structures, union.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
Y. Kanetkar, "Let Us C", BPB Publications, Sixteenth edition (2017).
B. S. Gottfried, "Programming with C", McGraw Hill Education, Forth edition (2018).
E Balagurusamy, "Computing Fundamentals and C Programming", McGraw Hill Education; Second edition (2017).

Reference Books:
1. P Dey and M. Ghosh, “Computer fundamentals and programming in C”, Oxford press, 2013.
2. Reema Thareja, “Computer fundamentals and programming in C”, Oxford press, 2013.
3. Schaum’s Outline, Programming with C.</td></tr>
</table>

THIRD SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	MAC331	Mathematics - III	3-1-0	4	4
2	CSC301	Discrete Mathematics	3-0-0	3	3
3	CSC302	Digital Logic Design	3-0-0	3	3
4	CSC303	Data Structures and Algorithms	3-1-0	4	4
5	PHC331	Physics of Semiconductor Devices	3-0-0	3	3
6	PHS381	Semiconductor Devices Laboratory	0-0-3	1.5	3
7	CSS351	Digital Logic Design Laboratory	0-0-3	1.5	3
8	CSS352	Data Structures and Algorithms Laboratory	0-0-4	2	4
9	XXS381	Co-curricular Activities - III (Optional)	0-0-0	0	0
		TOTAL	**15-2-10**	**22**	**27**

CSC 301 Discrete Mathematics 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 301</td><td>Discrete Mathematics</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Remember the basic terms, definitions and concepts of mathematics.
● CO2: Students will be able to understand the key concepts of discrete mathematics such as functional mapping, mathematical logic, counting principles, generating functions, algebraic structures and graph theory.
● CO3: Students will be able to apply the learned concepts to solve various problems.
● CO4: Students will be able to differentiate or relate the various ideas with respect to problems.
● CO5:Students will be able to judge the formulas and ideas to be applicable to a problem.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Set Theory: Definition of Sets, Venn Diagrams, complements, cartesian products, power sets, counting principle, cardinality and countability (Countable and Uncountable sets), proofs of some general identities on sets, pigeonhole principle. (3L)

Relation: Definition, types of relation (reflexive, symmetric, transitive, antisymmetric, Equivalence, partial ordering relations), composition of relations, domain and range of a relation, pictorial representation of relation, properties of relation, Partial Order, Lattice, Hasse Diagram. (6L)

Function: Definition and types of function, composition of functions, recursively defined functions, Surjection, Injection, Bijection, Composition of Function, Asymptotic notations: big-Oh, Theta, big-Omega. (4L)

Propositional logic: Proposition logic, basic logic, logical connectives, truth tables, tautologies, contradiction, normal forms (conjunctive and disjunctive), modus ponens and modus tollens, validity, predicate logic, universal and existential quantification. Notion of proof: proof by implication, converse, inverse, contrapositive, negation, and contradiction, direct proof, proof by using truth table, Proof by Well ordering principle. (6L)

Combinatorics: Mathematical induction, recursive mathematical definitions, basics of counting, permutations, combinations, inclusion-exclusion, recurrence relations (nth order recurrence relation with constant coefficients, Homogeneous recurrence relations), generating function (closed form expression, properties of G.F., solution of recurrence relation using G.F, solution of combinatorial problem using G.F.). (8L)

Algebraic Structure: Binary composition and its properties definition of algebraic structure; Semi group, Monoid, Groups, Abelian Group, properties of groups, Permutation Groups, Sub Group, Cyclic Group. (6L)</td></tr>
</table>

	Graphs: Graph terminology, types of graph, connected graphs, components of graph, Euler graph, Hamiltonian path and circuits, Degree Sequence, Radius, Diameter, Center of a graph, Graph coloring, Chromatic number. Planarity of a graph: K(3,3) and K(5). Clique, Independent set, bipartite graph, Tree: Definition, types of tree (rooted, binary), properties of trees. (9L)
Text Books, and/or reference material	**Text Books:** 1. C. L. Liu, Elements of Discrete Mathematics, Tata McGraw Hill. 2. Norman L. Biggs, Discrete Mathematics, Oxford. 3. Douglas B. West, Introduction to Graph Theory, Prentice Hall, India. **Reference Books:** 1. Ronald L. Graham, Donald E. Knuth and O. Patashnik, Concrete Mathematics, Pearson Education.

CSC302 Digital Logic Design 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 302</td><td>Digital Logic Design</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites:</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">At the completion of this course students will be able to:
CO1: Realize the various logic gates and and laws of Boolean algebra. Analyse different types of digital electronic circuit using various mapping and logical tools.
CO2: Design and analyses the various combinational circuits.
CO3: Design and analyses the various sequential circuits.
CO4: Design and analyse combinational and sequential logic circuits through HDL models.
CO5: Synthesis the various logic using ASM charts.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT-I: Switching Circuits, Various number system and their conversions: Arithmetic of these number systems, Complements, Data Representation: Binary numbers, binary codes, fixed point representation, floating point representation, Code and their conversions, Addition and Subtraction on Codes, Error Detection codes (Hamming code etc), representation of signed binary number in Fixed and Floating Points. (5L)

UNIT-II: Boolean algebra, logic gates, and switching functions, truth tables and switching expressions, minimization of completely and incompletely specified switching functions, Karnaugh map and Quine-McCluskey method, multiple output minimization, representation and manipulation of functions using BDDs, two-level and multi-level logic circuit synthesis. (10L)
UNIT-III: Combinational logic circuits: Realization of Boolean functions using NAND/NOR Gates, Decoders, multiplexers. Logic design using ROMs, PLAs and FPGAs. Case Studies. (8L)

UNIT-IV: Sequential circuits: Clocks, flip-flops, latches, counters and shift registers, finite-state machine model, synthesis of synchronous sequential circuits, minimization and state assignment, asynchronous sequential circuit synthesis. (12L)

UNIT-V: ASM charts: Representation of sequential circuits using ASM charts, synthesis of output and next state functions, data path control path partition-based Design. (7L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Digital Logic Design, M. Morris Mano, Michael D Cilleti, PHI.

Reference Books:
1. Digital Principles & Application, 5th Edition, Leach & Malvino, McGraw Hill Company.
2. Modern Digital Electronics, 2nd Edition, R.P. Jain. Tata Mc Graw Hill Company Limited.</td></tr>
</table>

CSC303 Data Structures and Algorithms 3-1-0 4 Credits 4 Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 303</td><td>Data Structures and Algorithms</td><td>PCR</td><td>3</td><td>1</td><td>0</td><td>4</td><td>4</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">CSC-01 (Introduction to Computing)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">CO1: Understanding the fundamental concepts of data, data types and abstract data types.
CO2: Implementation of different abstract data types using different data structures.
CO3: Apply different types of data structures to implement different application problems.
CO4: Analysis of the suitability/compatibility of different data structures based on the types of applications.
CO5: Design and development of algorithms for real-life applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to problem solving through computer, Design of algorithm to solve a problem, Concept of static and dynamic memory allocation, Algorithms and data structures, Concept of Abstract Data Type (ADT) with examples. (3L)

Efficiency of an algorithm, Asymptotic notations, Time and space complexities, Analysis of algorithms, Comparing asymptotic running times, Impact of data structure on the performance of an algorithm. (4L)

Array, Single and multi-dimensional array, Memory representation (row major and column major) of array, Insertion, and deletions in array, Advantages and disadvantages of array. (3L)

Linked list as an ADT, Memory allocation and deallocation for a linked list, Linked list versus array, Types of linked lists: singly linked list, doubly linked list and circular linked list, Operations on linked list: creation, display, insertion and deletion (in different positions), summation, average, maximum, minimum etc. Application of linked list: representations and operations on polynomials, sparse matrices. (7L)

Stack as an ADT, Main operations (push and pop), auxiliary operations and axioms, Array implementation of stack, Limitation of array implementation, Linked list implementation of stack, Applications of stack: Recursion, Function call, Evaluation of postfix expression using stack, Conversion of infix to postfix using stack. (6L)

Queue as an ADT, Main operations (enqueue and dequeue), Auxiliary operations and axioms, Array implementation of queue, Limitation of array implementation and Circular queue, Linked list implementation of queue, Double ended queue (dequeue) Priority queue and its applications. (5L)</td></tr>
</table>

	Binary Tree, Definition and properties, Representation of binary tree in memory: linked representation, array representation, Binary tree traversal, Preorder, Inorder and Postorder, Expression tree, Heap and its applications. (5L) Search trees: Binary search tree, Balanced binary search tree, AVL tree, Red Black tree, M-way tree, M-way search tree, B tree, B+ Tree. (7L) Searching: Linear search and binary search. (3L) Sorting: Bubble, selection, insertion, Quick sort, Merge sort, Heap sort, Radix sort. (7L) Graphs: Mathematical Properties, Degree, Connectedness, Representation using matrix, Adjacency list, Directed Graphs, Directed Acyclic Graph. (2L) Hashing: Hash functions. Collision, Collision resolution techniques: linear probing, quadratic probing, double hashing, chaining, Rehashing. (4L)
Text Books, and/or reference material	**Text Books:** 1. R. F. Gilberg and B. A. Forouzan, "Data Structures: A pseudocode approach with C", 2nd Edition, CENGAGE Learning. 2. A. V. Aho, J. D. Ullman and J. E. Hopcroft, “Data Structures and Algorithms”, Addition Wesley. 3. Lipschutz, “Data Structures (Schaum’s Outline Series)”, Tata Mcgraw Hill. 4. E. Horowitz, S. Sahni, S. Anderson-Freed, “Fundamentals of Data Structures in C”, Universities Press; Second edition (2008). **Reference Books:** 1. Y. Langsam, M. J. Augenstein and A. N. Tanenbaum, "Data Structures using C and C++", Pearson, 2006. 2. Knuth, Donald E. The Art of Computer Programming. 3rd ed. Vols 1&2. Reading, MA: Addison-Wesley, 1997. ISBN: 0201896834. ISBN: 0201896842. ISBN: 0201896850. 3. Kleinberg and Eva Tardos. Algorithm Design. Addison-Wesley 2005 ISBN-13: 978-0321295354.

CSS351 Digital Logic Design Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><th colspan="7">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS351</td><td>Digital Logic Design Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="4">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">NIL</td><td colspan="4">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Understand basic gate operations.
● CO2: Realize the boolean function using basic gates in both SOP/POS form.
● CO3: Realize different combinational circuits with basic gates.
● CO4: Understand the basic structure of different digital components- multiplexer, decoder, encoder etc.
● CO5: Verification of state table of different flip flop using NAND/NOR gate.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1. Familiarization with IC, study of the data sheet, VCC, Ground. Verification of the truth tables.
2. Implementation of a given Boolean function using logic gates in both SOP and POS forms. Verify the Universal logic gate (NAND, NOR).
3. Verify DE Morgan’s law.
4. Implement NAND based logic circuit for any Boolean expression. Verify that a Boolean expression, e.g. F= AB + A’C’. is functionally complete.
5. Implement a Full adder using Half Adder. Implement the combinational circuit to realize both Adder and Subtractor together.
6. Implementation and verification of Decoder, Multiplexer, Encoder and Priority Encoder etc.
7. Implement and verify Ripple Carry Adder, Carry Look Ahead Adder and BCD Adder.
8. Verification of state tables of RS, JK, T and D flip-flops using NAND & NOR gates.
9. Implement and verify the 4-bit counter</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books:
1. Digital Logic Design, M. Morris Mano, Michael D Cilleti, PHI.

Others:
1. Laboratory Manual.</td></tr>
</table>

CSS352 Data Structures and Algorithms Laboratory 0-0-4 2 Credits 4Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS352</td><td>Data Structures and Algorithms Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>4</td><td>4</td><td>2</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2">CSC-01 (Introduction to Computing), CSS-51 (Computing Laboratory)</td><td colspan="5">CT+EA [CT: 60%, EA (Programming assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Choose appropriate data structures for representation and manipulation of the data for the given problems.
● CO2: Handle operations like search, insertion, deletion, traversing and sorting on various data structures.
● CO3: Have knowledge on the applications of linear and non-linear data structures for real life problems.
● CO4: Able to store and manipulate data in an efficient manner.
● CO5: Able to implement stack, queue, binary tree, etc. using arrays and linked lists.
● CO6: Able to apply the concepts learnt through this course in various domains like DBMS and compiler.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1. Insertion and deletion in arrays using dynamic memory allocation.
2. Linear search, Binary search (recursive, non-recursive).
3. Memory allocation and deallocation for linked list.
4. Operations on linked list: creation, display, insertion and deletion (in different positions), summation, average, maximum, minimum etc.
5. Array implementation of stack and queue.
6. Linked implementation of stack and queue.
7. Evaluation of postfix expression using stack.
8. Conversion of infix expression to its postfix version using stack.
9. Linked implementation of binary tree and preorder, inorder and postorder traversal on binary tree.
10. Implementation of binary search tree and operations on it (searching, insertion, deletion).
11. Implementation of height-balanced binary search tree (AVL tree).
12. Implementation of 2-3 tree.
13. Implementation of Chaining.
14. Implementation of sorting algorithms: Selection sort, insertion sort, bubble sort, quick sort, heap sort, merge sort, radix sort.
15. Implementation of few basic graph operations (such as breadth first and depth first traversal, finding minimum spanning tree, shortest path) on graph.</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. S. Lipschutz, “Data Structures (Schaum’s Outline Series)”, McGraw Hill Education; First edition (2017). 2. E. Horowitz, S. Sahni, S. Anderson-Freed, “Fundamentals of Data Structures in C”, Universities Press; Second edition (2008). 3. E. Balagurusamy, "Programming in ANSI C", McGraw Hill Education India Private Limited, Seventh edition (2017). **Reference Books:** 1. B. S. Gottfried, "Programming with C", McGraw Hill Education, Fourth edition (2018).

FOURTH SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	CSC401	Computer Organization and Architecture	3-1-0	4	4
2	CSC402	Theory of Computation	3-0-0	3	3
3	CSC403	Design and Analysis of Algorithms	3-1-0	4	4
4	CSC404	Object Oriented Programming	3-0-0	3	3
5	CSC405	Signals and Systems	3-0-0	3	3
6	YYO44*/ HSC431	Open Elective - 1/ Psychology	3-0-0	3	3
7	CSS451	Computer Organization Laboratory	0-0-3	1.5	3
8	CSS452	Object Oriented Programming Laboratory	0-0-3	1.5	3
9	CSS453	Signal Processing Laboratory	0-0-3	1.5	3
10	XXS481	Co-curricular Activities - IV (Optional)	0-0-0	0	0
		TOTAL	**18-2-19**	**24.5**	**29**

4th Semester

	Basket of Open Elective – 1
CSO441	Data Structures and Algorithms
CSO442	Object Oriented Technology

CSC 401 Computer Organization and Architecture 3-1-0 4Credits 4Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 401</td><td>Computer Organization and Architecture</td><td>PCR</td><td>3</td><td>1</td><td>0</td><td>4</td><td>4</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Digital Logic Design (CSC302)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Analyze the various parts of a modern computer functional units, bus structure, addressing modes and Computer arithmetic.
● CO2: Identify the process involved in executing an instruction and fetching the word from memory.
● CO3: Design the hardwired and micro-programmed control units and implementation of interrupts.
● CO4: Understand the memory hierarchy and design a memory system.
● CO5: Understand Pipelined execution and instruction scheduling.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT-I: Introduction: Evolution of computers, Basic Structure of Computers: Basic Operational Concepts, GPR based and stack based organisation. Bus Structures, Performance Measurement: Processor Clock, Basic Performance Equation, Clock Rate, Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Encoding of Machine Instructions (Huffman encoding etc). (12L)

UNIT-II: Fundamental concepts of the processing Unit: Fetching and Storing words, Register Transfer, Execution of instruction, Arithmetic Operations: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Combinational and Sequential ALU, ALU expansion strategies, Design of Multipliers and Dividers, Wallace tree and Booth's Multipliers, Floating Point Numbers (IEEE754), Floating Point Operations, Multiplication of Positive Numbers, Signed Operand Multiplication (Booth's Multiplication etc.), Fast Multiplication, Integer Division. (10L)

UNIT-III: Computer Organization and Design (Datapath and control path): Instruction codes, computer registers, computer instructions, timing & control, instruction cycle, memory reference instructions, Hard-wired Control, Micro programmed Control: Micro instruction, Microprogram sequencing, Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, Bus Arbitration schemes, USB. (Brief overview of 8085/8086 microprocessor). (12L)

UNIT-IV: Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, page mode access, interleaved access. Performance Considerations, Virtual Memories, Secondary Storage. (12L)</td></tr>
</table>

	UNIT-V: Basic concepts of pipelining, the instruction pipeline – pipeline hazards – instruction level parallelism – reduced instruction set –Computer principles – RISC versus CISC. Introduction to GPP, ASIP and ASIC etc. (10L)
Text Books, and/or reference material	**Text Books:** 1. David A Patterson, John L Hennessy, "Computer Organization and Design", (The Hardware/Software Interface) Morgan Kaufmann. 2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill. **Reference Books:** 1. William Stallings, "Computer Organization and Architecture". 2. Nicholas P Carter, "Computer Architecture & Organisation".

CSC 402 Theory of Computation 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC402</td><td>Theory of Computation</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Discrete Mathematics (CSC 301)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Explain the concept of regular languages through regular expressions and finite automata.
● CO2: Describe context-free languages and context free grammars.
● CO3: Design grammars and automata for various languages.
● CO4: Examine the power of Turing machines and design TM for simple problems.
● CO5: Analyze the concept of undecidability in the context of Turing machine design.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Regular sets and Regular Expression, Non-deterministic and deterministic finite automata and their equivalence, Minimization of deterministic finite automata, Regular expressions to Finite Automata. (10L)
2. Finite Automata with outputs. (2L)
3. Properties of Regular Sets: Pumping Lemma, Closure Properties, Decision algorithms. (5L)
4. Context Free Grammars. Derivations. Ambiguity in grammars. (3L)
5. Chomsky hierarchy of languages and grammars. Regular grammars. (3L)
6. Normal Forms for Context free grammars. CNF and GNF. Closure properties of context free languages, Pumping lemma for context free languages. Decision Properties. (10L)
7. Pushdown automata. (3L)
8. Turing machines. Unrestricted Grammars. Properties of recursive and r.e.languages, Undecidability. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Introduction to Automata Theory, Languages and Computation by J.E.Hopcroft, Rajiv Motwani and J.M.Ullman. Pearson Education.
2. Introduction to Languages and Theory of Computation By John C. Martin McGraw Hill Education

Reference Books:
1. Elements of the Theory of Computation By Harry R. Lewis and Christos H. Papadimitriou Prentice Hall of India.
2. Theory of Automata and Formal Languages By Anand Sharma University Science Press</td></tr>
</table>

CSC 403 Design and Analysis of Algorithms 3-1-0 4Credits 4Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 403</td><td>Design and Analysis of Algorithms</td><td>PCR</td><td>3</td><td>1</td><td>0</td><td>4</td><td>4</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Discrete Mathematics (CSC 301), Data Structure and algorithm (CSC 303)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Students will be able to define many important concepts such as asymptotic analysis, dynamic programming, recurrences etc.
● CO2: Students will be able to describe the key ideas of different algorithm design paradigms.
● CO3: Can apply different algorithmic ideas efficiently to solve new problems.
● CO4: Students can analyze and understand the time complexity of the algorithms, and its correctness.
● CO5: Can evaluate the hardness of an algorithm if required.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction and basic concepts: Algorithm, Asymptotic notations (big-Oh, big Omega, Theta, small-oh) and their significance, introduction to RAM model of computation, complexity (Time Complexity, Space Complexity) analysis of algorithms, worst case and average case. Solving Recurrences – Substitution method, Recurrence tree method and Master Method, Finding maximum and minimum of n numbers, Finding the second largest of n numbers and exact number of comparisons. (7L)

Lower bound: Lower bound for a problem. Computing the lower bound for sorting (comparison based sorting) and computing the lower bound for computing convex hull using the lower bound for sorting problem. (2L)

Amortized complexity analysis: aggregate analysis, accounting method and potential method. Examples: storage allocation problem, binary counting problem and heap sort. (4L)

Using Induction to Design algorithm: The celebrity problem, Majority Finding problem. (2L)

Divide and conquer Problem: Multiplication of two n-bit integers, Strassen’s Matrix Multiplication problem, Closest pair of points, linear time median finding algorithm, Convex hull and its computation. (6L)

The Greedy Algorithm: Greedy algorithms and their correctness proof: Interval scheduling problem, Interval partitioning problem, Minimizing the Lateness of Intervals problem, Fractional Knapsack Problem. (5L)

Dynamic Programming: Longest Common Subsequence, Matrix Chain Multiplication, 0-1 Knapsack Problem, longest common subsequence problem. (6L)</td></tr>
</table>

	Backtracking Method, Branch and Bound Method. (2L) **Graph Algorithms:** Depth First Search, Breadth First Search, Dijkstra's Single Source Shortest Path algorithm; All pair shortest path algorithm, Minimum Spanning Tree (Prim's and Kruskal's algorithm). (7L) **Randomized Algorithm**: Las Vegas and Monte Carlo; Randomized Quick Sort algorithm and Min Cut problem. (3L) **Reducibility between problems and NP-completeness:** Different class of Problems (P, NP, NP-Hard, NP-Complete), Discussion of different NP-complete problems like satisfiability, clique, vertex cover, independent set, Hamiltonian cycle, set cover, dominating set problem. (6L) **Approximation Algorithm:** Approximation ratio for maximization problem and minimization problem, Constant ratio approximation algorithms for metric travelling salesperson problem (TSP) and vertex cover problem, log n ratio approximation algorithm for Set Cover problem. (6L)
Text Books, and/or reference material	**Text Books:** 1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, Introduction to Algorithms, by Prentice Hall India. 2. J. Kleinberg and Eva Tardo, Algorithm Design by Pearson Education (Indian edition). 3. S. Dasgupta, C. Papadimitriou and U. Vazirani, Algorithms, by Tata McGraw-Hill. **Reference Books:** 1. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006. 2. Algorithms: Design Techniques and Analysis Volume 7 of Lecture notes series on computing, World Scientific, 1999. **Others:** Tim Roughgarden's video lectures and notes of CS161 and *CS261; NPTEL's lectures on Design and Analysis of Algorithms; NMEICT video on Design of Algorithms (http://www.nmeict.iitkgp.ac.in/Home/videoLink/10/3gp)*.

CSC 404 Object Oriented Programming 2-1-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 404</td><td>Object Oriented Programming</td><td>PCR</td><td>2</td><td>1</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Introduction to Computing (CSC01), Data Structures and Algorithms (CSC303)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Apply Object oriented approach to design software.
● CO2: Implement programs using classes and objects.
● CO3: Specify the forms of inheritance and use them in programs.
● CO4: Analyze polymorphic behavior of objects.
● CO5: Design and develop GUI programs.
● CO6: Develop Applets for web applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Course Introduction- Concepts of Object Oriented Programming, Procedural approach, Limitation of Procedural Language, Object concept. (2L)

Object Oriented Terminologies- Class concept, ADT, encapsulation, Cardinality, Data hiding, Inheritance, Polymorphism, Advantages of OOPs, Advantages of OOPs, difference between Procedural and Object Oriented Language, Evolution of C++. (4L)

Basic Input/Output in C++ - The 1st C++ Program (temperature conversion), compilation, Input stream and output stream, Advantages of cin a cout over pritnf and scanf. (3L)

Basic C++ features - Literals, Constants, Manipulators, Assertions, Enumerated Data Types, Scope resolution operator. (5L)

Pointers & References in C++- Basic operations on pointers, Array of pointers, pointer to an array, self referential structures, References in C++ , use of references. (7L)

Dynamic memory allocation/deallocation- Use of new and delete operator, multi-dimensional array allocation, Examples. (5L)

Constructor and Destructor, Various examples of constructors, Constructor Salient Features, Destructors,, Examples. (2L)

Functions in C++; Overloading- function call, Macros, and it’s limitations, Inline function, Function Overloading, Constructor Overloading, Examples, Function with Default arguments, Various Examples of Default arguments. (5L)

Writing C++ Classes- Class, C++ class vs Structure, This pointer, Memory Layout of C++ program, Static member of class. Static Member Functions, Static Object, Examples. (5L)

C++ Constants Revisited - Storage Allocation, Constants and References, Constant member data and Functions, Constants Objects, Examples. (2L)</td></tr>
</table>

	Friend Function & Operator Overloading - Friend Functions, Use of friend functions, friends as bridges, Various examples, Operator Overloading, examples, advantages of friend functions during overloading. (6L) **Templates in C++**, Generic function and classes, examples, syntax of a template, Template class (5L) **Inheritance in C++**, Derive class, Parameterized constructor in derive class, Protector Specifier, Examples of different types of inheritance, Virtual Base Class, Up casting. Polymorphism and virtual function, Function call finding, Virtual Functions, Examples. Lecture (38): V Table and V pointer, Pure Virtual Function, Examples. Lecture (39-40): Exception Handling in C++ Lecture (41): Unformatted Input/ Output operations, Formatted I/O functions, File handling. 2-3 Lectures are planned for doubt clearance.
Text Books, and/or reference material	**Text Books:** 1. Adam Drosdek, "DATA STRUCTURES AND ALGORITHMS IN C++" , Brooks/Cole Thomson Learning. 2. Bjarne Stroustrup "The C++ Programming Language", Pearson Education. 3. E. Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw Hill. **Reference Books:** 1. Bruce Eckel, "Thinking in C++", Prentice Hall. 2. S. B. Lippman, J. Lajoie, B. E. Moo, "C++ Primer", Addison-Wesley Professional 3. Bjarne Stroustrup, "Programming: Principles and Practice Using C++", Addison-Wesley Professional. 4. Effective C++: 50 Specific Ways to Improve Your Programs and Design by Scott Meyers, 1997. 5. More Effective C++ by Scott Meyers, 2002. **Others**: NPTEL course link by Prof. Partha Pratim Das - https://onlinecourses-archive.nptel.ac.in/noc19_cs10/preview

CSC 405 Signals and Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 405</td><td>Signals and Systems</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Calculus, Linear algebra</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">The students, after successfully completing the course, will be able to:
● CO1: Understand the definitions, classifications, properties and applications of signals and systems.
● CO2: Understand Laplace Transform, Fourier Transform, z-transform and other mathematical operations on signals.
● CO3: Understand Properties of LTI system.
● CO4: Analyze continuous time and discrete time signals in time and Transform domain by applying mathematical tools like Laplace transform, Fourier transform, z-transform.
● CO5: Design and analyze continuous and discrete time systems.
● CO6: Compare continuous time and discrete time systems in real life applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Signals and systems, introduction to signals, classification of signals; mathematical operations of signals, some standard signals, generating signals using standard signals. (6L)

Introduction to systems, classification of systems, Linear Time Invariant (LTI) Systems (continuous-time and discrete-time systems), properties of LTI systems, impulse response, convolution, causality, stability; (6L)
Impulse response of discrete-time LTI systems, discrete time convolution, difference equations and analysis, developing equivalent discrete-time system from a given continuous-time system and analysis of their stability; (4L)

Laplace Transform, Properties of Laplace Transform, Inverse Laplace Transform; (4L)

Applications of Laplace Transforms to design and analyse continuous-time systems, transfer function of continuous-time systems, poles and zeros, stability analysis; (4L)

Introduction to z-Transform, Properties of z-Transform, Region of Convergence, Inverse z-Transform; (3L)

Applications of z-Transforms to design and analyse Discrete Time Systems (3L)

Introduction to Fourier analysis, Fourier series for periodic signals, discrete spectrum of periodic signals; (2L)</td></tr>
</table>

	Introduction to Fourier transform, properties of Fourier transform, energy and power spectral density, frequency response of continuous-time systems, some problem examples; (4L) Fourier analysis of Discrete Signals, Discrete Time Fourier Transform (DTFT), Properties of DTFT, Examples of DTFT, DFT. (4L) Concept of state, state space analysis, state space representation of continuous time systems (2L)
Text Books, and/or reference material	**Text Books:** 1. Signals and Systems, 2nd ed., Simon Heykin and Barry Van Veen, John Wiley & Sons. 2. Signals and Systems, Oppenheim and Willsky, Prentice Hall Signal Processing Series. **Reference Books:** 1. Signal Processing and linear systems, B. P. Lathi, Oxford University Press. 2. Theory and Problems of Signals and Systems, Hsu, Schaum's Outline Series.

CSS 451 Computer Organization Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 451</td><td>Computer Organization Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="4">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">Digital Logic Design (CSC302), Digital Logic Design laboratory (CSS351)</td><td colspan="4">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Understand the basic structure of digital computer.
● CO2: Understand the synchronous / asynchronous logic.
● CO3: Perform different operations with flip-flop.
● CO4: Understand arithmetic and control unit operation.
● CO5: Understand the basic concepts of Memory.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1. Introduction to Verilog HDL and Implementation of basic logic gates using Verilog.
2. Familiarization of Assembly language programming.
3. Implementation of combinational circuits using Verilog.
4. Implementation of sequential circuits using Verilog.
5. Implementation of Booth’s Multiplier circuit.
6. Synthesis of simple data path and Controllers, Processor Design
7. Implementation of Random Access Memory (RAM) to perform both R/W operation.
8. Mini project.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books:
1. David A Patterson, John L Hennessy, “Computer Organization and Design”, (The Hardware/Software Interface) Morgan Kaufmann.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill.

Reference Books:
1. William Stallings, “Computer Organization and Architecture”.
2. Nicholas P. Carter, “Computer Architecture & Organisation”.

Others: Laboratory Manual</td></tr>
</table>

CSS 452 Object Oriented Programming Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS452</td><td>Object Oriented Programming Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">Introduction to Computing (CSC01), Data Structures and Algorithms (CSC303)</td><td colspan="5">CT+EA [CT: 60%, ET(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Design and develop utilities for screen and keyboard processing.
● CO2: Design and implement TSR programs.
● CO3: Develop programs using objects and inheritance in Java Language.
● CO4: Design and implement GUI programs using components in Java Language.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Assignment 1: Design codes using OOL syntax; use of manipulators, dynamic allocation, multi-dimensional array writing application like addition, subtraction, multiplication, finding factorial of a large numbers etc.
Assignment 2: Develop codes involving binary and text files involving string manipulation, graph processing, etc.
Assignment 3: Design class library for implementing matrix, complex number, string, stack, queue, linked list, heap, binary search tree, polynomial, etc.
Assignment 4: Develop class library to implement application like hashing, huffman code, expression evaluation using the libraries developed in assignment 3.
Assignment 5: Enhance the class libraries in assignment 3&4 implementing function overloading.
Assignment 6: Enhance the class libraries in assignment 3&4 implementing operator overloading.
Assignment 7: Develop codes using inheritance.
Assignment 8: Design and develop template classes.
Assignment 9: Implement exception handling in some existing template classes .</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Bruce Eckel, “Thinking in C++”, Prentice Hall.
2. S. B. Lippman, J. Lajoie, B. E. Moo, “C++ Primer”, Addison-Wesley Professional
3. Bjarne Stroustrup, “Programming: Principles and Practice Using C++”, Addison-Wesley Professional.
4. Effective C++: 50 Specific Ways to Improve Your Programs and Design by Scott Meyers, 1997.
5. More Effective C++ by Scott Meyers, 2002.

Others:

NPTEL course link by Prof. Partha Pratim Das - https://onlinecourses-archive.nptel.ac.in/noc19_cs10/preview</td></tr>
</table>

CSS 453 Signal Processing Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 453</td><td>Signals and Systems Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="2">Pre-requisites:</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2">MATLAB, Python</td><td colspan="5">CT+EA [CT: 60%, ET(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Simulate signals and systems using modern computer software packages (Matlab/Python).
● CO2: Generate and characterize various continuous and discrete time signals.
● CO3: Analyze signals time and frequency domain.
● CO4: Characterize system dynamics using impulse responses, transfer functions.
● CO5: Analyze the systems using Laplace transform and Z-transform.
● CO6: Design and analyze linear time-invariant (LTI) systems.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1. Introduction to Computer Software Package Matlab/Python
2. Simulation of standard of signals like
a. Unit step
b. Unit impulse
c. Ramp
d. Periodic sinusoidal sequences.
3. Basic operation on signals: Addition, Subtraction, Multiplication, Division, shifting, scaling, etc.
4. Convolve and analyze signals in time domain.
5. Laplace transform and inverse Laplace transform of signals.
6. Convolution of signals in transformed domain and verification of convolution property of Fourier and Z-transform.
7. Study of LTI system and its stability.
8. Design of Stable LTI systems.
9. Design of FIR and IIR systems.
10. Implement Fast Fourier Transform algorithm of a signal.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books:
1. Signals and Systems Laboratory with MATLAB, Alex Palamides and Anastasia Veloni, CRC Press, 2011.
Reference Books:
2. Anywhere-Anytime Signals and Systems Laboratory, Nasser Kehtarnavaz, Fatemeh Saki, Morgan & Claypool, 2017.</td></tr>
</table>

Open Elective – 1

CSO441 Data Structures and Algorithms 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 441</td><td>Data Structures and Algorithms</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">CSC-01 (Introduction to Computing)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding the fundamental concepts of data, data types and abstract data types.
● CO2: Implementation of different abstract data types using different data structures.
● CO3: Design and development of algorithms for real-life applications.
● CO4: Apply different types of data structures to implement different algorithms.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to problem solving through computer, Design of algorithm to solve a problem, Concept of static and dynamic memory allocation, Algorithms and data structures, Concept of Abstract Data Type (ADT) with examples. (4L)

Efficiency of an algorithm, Time and space complexities, Impact of data structure on the performance of an algorithm. (3L)

Array, Single and multi-dimensional array, Memory representation (row major and column major) of array, Insertion, and deletions in array, Advantages and disadvantages of array. (3L)

Linked list as an ADT, Memory allocation and deallocation for a linked list, Linked list versus arrays, Types of linked lists: singly linked list, doubly linked list, circular linked list, Operations on linked list: creation, display, insertion and deletion (in different positions). (5L)

Stack as an ADT, Main operations (push and pop), auxiliary operations and axioms, Array implementation of stack, Limitation of array implementation, Linked list implementation of stack, Applications of stack: Recursion, Function call, Evaluation of postfix expression using stack. Conversion from infix expression to its postfix version. (5L)

Queue as an ADT, Main operations (enqueue and dequeue), Auxiliary operations and axioms, Array implementation of queue, Limitation of array implementation and Circular queue, Linked list implementation of queue, Priority queue and its applications. (4L)

Trees, Definition and mathematical properties, Binary trees, Representation of binary trees in memory: linked representation, array representation, Binary tree traversal, Pre-order, Inorder, Post order, Expression trees, Heap and its applications, Search trees: Binary search trees, Balanced binary search trees. (8L)</td></tr>
</table>

	Searching and sorting: Linear search and binary search, Bubble, selection, insertion, Quick sort, Merge sort, Heap sort, Radix sort. (8L) Graphs: Mathematical Properties, Degree, Connectedness, Memory representation of graph: adjacency matrix, Adjacency list, Directed Graphs, Directed Acyclic Graph. (2L)
Text Books, and/or reference material	**Text Books:** 1. Lipschutz, "Data Structures (Schaum's Outline Series)", Tata Mcgraw Hill. 2. E. Horowitz, S. Sahni, S. Anderson-Freed, "Fundamentals of Data Structures in C", Universities Press; Second edition (2008). **Reference Books:** 1. Y. Langsam, M. J. Augenstein and A. N. Tanenbaum, "Data Structures using C and C++", Pearson, 2006. 2. Kleinberg and Eva Tardos. Algorithm Design. Addison-Wesley 2005 ISBN-13: 978-0321295354.

CSO442 Object Oriented Technology 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 442</td><td>Object Oriented Technology</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">CSO442 (Object Oriented Technology)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: Apply Object oriented approach to design software.
• CO2: Implement programs using classes and objects.
• CO3: Specify the forms of inheritance and use them in programs.
• CO4: Analyze polymorphic behavior of objects.
• CO5: Design and develop GUI programs.
• CO6: Develop Applets for web applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to problem solving through computer, Design of algorithm to solve a problem, Concepts of functions, loops, strings, arrays, pointers, structures etc. Procedure Oriented Programming, Object Oriented Programming, Objects and Classes, 3 basic features of OOP, Comparison of procedural programming and object oriented programming, C++ language, cout, cin operator, return type of main, structure of a C++ program, example with description, Tokens, keywords, identifiers, declaration of variables, dynamic initialization of variables, reference variables, scope resolution operator, difference between C and C++. Examples and Practice Sessions. (7L)

Declaration of classes and objects, member functions, accessing class members, inline function, Nesting of member function, Private member function, Static data members, static member function, Objects as function argument, Friend functions, structure and class, returning objects, Examples and Exercises. (5L)

Overview of constructors, default constructors, parameterized constructors, constructors with default arguments, dynamic initialization of objects, copy constructors, dynamic constructors & destructors, constraints on constructors & destructors. Examples and Exercises. (4L)

Operator overloading overview, defining operator overloading function, Overloading unary operator, binary operators and arithmetic operators, Overloading using friend function, multiple overloading, Overloading comparison operators, conversion between objects and basic types, conversion between objects of different classes, overloading various operators, such as +, -, *, /, =, ==, (), [], {}, &&, ||, ++ (preincrement and post increment) etc. Examples and Exercises. (6L)

Overview, defining derived classes, types of inheritance, single inheritance, making private member inheritable, multilevel inheritance, Multiple inheritance, ambiguity in multiple inheritance Hierarchical inheritance,hybrid inheritance, Virtual base classes, abstract classes, Constructors in derived classes, initialisation list, nesting of classes, Examples and Exercises. (7L)</td></tr>
</table>

	Overview, late binding, early binding, Pointers to objects, accessing class members using pointers, creating objects at runtime, This pointer, pointers to derived classes, virtual functions, pure virtual functions, Examples and Exercises. (5L) Overview of Templates, generic class, function template, function template with multiple argument, Class template, Class template with multiple argument, overloading template function, templates as member function of a class, Examples and Exercises. (3L) Exception handling overview, exception handling mechanism, throwing and catching mechanism, Multiple catch, catch All exceptions, rethrowing an exception, Examples and Exercises. (3L) Mini Project Implementation using the concepts. (2L)
Text Books, and/or reference material	**Text Books:** 1. E Balagurusamy, "Object oriented programming in C++", Mc Graw Hill, ISBN 978-93-5260-779-0. 2. Herbert Schildt, "Teach yourself C++", Mc Graw Hill, 3rd Edition, ISBN 0-07-882311-0. 3. Herbert Schildt, "C++: The Complete Reference", Mc Graw Hill, 4th Edition, ISBN 0-07-212124-6. **Reference Books:** 1. Stroustrup, "The C++ Programming Language", 3rd Edition, 2002,Addison Wesley. 2. Eckel, "Thinking in C++", Vol1, 2nd Edition, 2002, Pearson. 3. R. Lafore, "Object Oriented Programming with C++", 4th Edition, 2008, Pearson.

FIFTH SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	CSC501	Operating Systems	3-0-0	3	3
2	CSC502	Database Management System	3-1-0	4	4
3	CSC503	Compiler Design	3-0-0	3	3
4	CSC504	Microcontroller based Systems	3-0-0	3	3
5	YYO54*	Open Elective - 2	3-0-0	3	3
6	CSS551	Design and Analysis of Algorithms Laboratory	0-0-3	1.5	3
7	CSS552	Microcontroller based System Laboratory	0-0-3	1.5	3
8	CSS553	Operating Systems Laboratory	0-0-3	1.5	3
9	XXS581	Co-curricular Activities - V (Optional)	0-0-0	0	0
		TOTAL	**15-1-9**	**20.5**	**25**

5th Semester

	Basket of Open Elective – 2
CSO541	**Fundamentals of Algorithms**
CSO542	**Database Management System**
CSO543	**Computer Organization**
CSO544	**Operating Systems**

CSC 501 Operating Systems **3-0-0** **3Credits** **3Hours**

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 501</td><td>Operating Systems</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Computer Organization and Architecture (CSC401), Introduction to Computing (CSC01), Data Structures and Algorithms (CSC303)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding functional architecture of an operating system.
● CO2: Develop algorithms for subsystem components.
● CO3: Design device drivers and multi-threading libraries for a tiny OS.
● CO4: Develop application programs using UNIX system calls.
● CO5: Design and solve synchronization problems.
● CO6: Understand standard UNIX and FAT file systems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introductory Concepts: Introduction to Operating System as a whole, memory, CPU(registers and ALU), Evolution of Operating System-types of OS(advantages and drawbacks), Performance measurement metrics. (5L)

Process Data Structures and State transitions: Process management, Basic Definitions, Process table, PCB(process control block), PTE(process table entry), Process states, Transition diagram, context of process-user level, kernel-level and process Level. (4L)

Process Control: Process creation, Parent and Child processes, System calls--fork(), exit(), wait(), kill(), Signal handling, Process scheduling strategies-FCFS, SPN, SRT, Round Robin, HRRN, Fair share scheduling. (6L)

Multi-threading: Threads in OS, thread vs process, ULT & KLT, Applications of threads, Use of POSIX threads library.
(4L)

Process synchronization - Race condition, Critical section, Process Sync Solution using Algorithmic approach (Lamport bakery Algorithm), Creating shared memory using POSIX library. (3L)

Semaphore- Binary and Counting semaphore, P() and V() operations, Solving Classical problem using semaphores- Sleeping barber, Producer-consumer, Reader-writer, Dining philosophers’s problem, Posix library for semaphores. (7L)

Monitors - Solving Classical problems using monitors. (4L)

Deadlocks - Necessary and sufficient conditions for deadlocks, approaches to deal</td></tr>
</table>

	with deadlocks, Deadlock Prevention, Avoidance (Banker's algorithm) and Detection. (3L) **Memory organization & management** - Virtual memory organization, Pure Paging, Pure Segmentation, Combined Paging-Segmentation, Inverted PMT, Page fault handling algorithms, Working set theory. (7L) **File management**- Directory structure, Storage of files on disks, contiguous and non-contiguous file allocation strategies, Internal and external fragmentation, FAT & Inode Structure, Free Space management, Disk scheduling strategies. (6L) **I/O management concepts** (2L)
Text Books, and/or reference material	**Text Books:** 1. "Operating System Concepts", Silberschatz and Galvin. 2. "Operating Systems: Internals and Design Principles" by William Stalling. 3. "Operating Systems: A Concept-Based Approach" by D M Dhamdhere. **Reference Books:** 1. "Operating System: A Design-oriented Approach" by Charles Crowley. 2. "Operating Systems: A Modern Perspective" by Gary J Nutt. 3. "Design of the Unix Operating Systems" by Maurice Bach. 4. "MODERN OPERATING SYSTEMS" by Andrew S Tanenbaum. **Others:** ● https://nptel.ac.in/courses/106/106/106106144/# Course "Introduction to Operating Systems" by PROF. CHESTER REBERIO, IIT Madras. ● https://nptel.ac.in/courses/106105214/ Course "Operating System Fundamentals" by Prof. Santunu Chattopadhyay, IIT Kharagpur.

CSC 502 Database Management System 3-1-0 4Credits 4Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 502</td><td>Database Management System</td><td>PCR</td><td>3</td><td>1</td><td>0</td><td>4</td><td>4</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Programming knowledge, Data Structures and Algorithms</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the basic concepts and appreciate the applications of database systems.
● CO2: Comprehend the fundamentals of design principles for logical design of relational databases.
● CO3: Apply the query writing skill and its subsequent optimization.
● CO4: Discuss the basic issues of transaction processing and concurrency control.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Concept & Overview of DBMS, Applications, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS. (4L)

Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features. (5L)

Relational Model: Structure of relational Databases, Relational Algebra, Relational Calculus, Extended Relational Algebra Operations, Views, Modifications of the Database. (7L)

SQL and Integrity Constraints: Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints, assertions, views, Nested Subqueries, Database security application development using SQL, Stored procedures and triggers. (7L)

Index Structures: Necessity of index structures, Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes, Dynamic Multilevel Indexes using B tree and B+ tree . (4L)

Normalization: Functional Dependency, Anomalies in a Database, The normalization process: Conversion to first normal form, Conversion to second normal form, Conversion to third normal form and BCNF, Fourth Normal form and fifth normal form, normalization and database design, Denormalization, Loss-less join decomposition, Dependency preservation. (8L)

Transaction processing: Introduction of transaction processing, advantages and disadvantages of transaction processing system, online transaction processing system, serializability and recoverability, view serializability. (5L)</td></tr>
</table>

	Concurrency Control: Serializability: Enforcing, Serializability by Locks, Locking Systems With Several, Lock Modes, Architecture for a Locking Scheduler Managing Hierarchies of Database Elements, Concurrency Control by Timestamps, Concurrency Control by Validation. (5L) **Database recovery management:** Deferred database modification Vs. Immediate database modification, Check point technique. (3L) **Query Optimization:** Heuristics in Query Optimization, Converting Query Tree to Query Evaluation Plan. (4L) **Distributed Database (DDB):** Introduction of DDB, DDBMS architectures, Homogeneous and Heterogeneous databases, Distributed data storage, Advantages of Data Distribution, Disadvantages of Data Distribution Distributed transactions, Commit protocols, Data Replication, Data Fragmentation. Distributed database transparency features. (4L)
Text Books, and/or reference material	**Text Books:** 1. "An Introduction to Database Systems", C. J Date, Pearson Education. 2. "DatabaseSystem Concepts", Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw-Hill. 3. "Distributed Databases Principles & Systems", Stefano Ceri and Giuseppe Pelagatti, McGraw-Hill International Editions. **Reference Books:** 1. "Fundamentals of Database Systems", Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley. **Others:** https://onlinecourses-archive.nptel.ac.in/noc18_cs15/preview

CSC 503 Compiler Design 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 503</td><td>Compiler Design</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites
Theory of Computing/
Theory of Automata</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: Idea of the difference between Compiler and other various Translators, Phases of a Compiler and Bootstrapping.
• CO2: Understand Lexical Analyzer, Transition Diagram of different tokens, Reserved Word Strategy.
• CO3: Idea of Syntax Analyzer, Ambiguity, Parse Tree, Top Down and Bottom Up Parser.
• CO4: Concept of Semantic Analyzer, Semantic Actions, Intermediate Code, Virtual Machine. Lexical and Grammatical Errors.
• CO5: Idea of Code Optimization, Criterion of Optimization, Different Local and Global Optimization Techniques.
• CO6: Idea of Code Generation, Instruction Costs, Code Generation Algorithm, Run Time Store Management.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Idea of the difference between Compiler and other various Translators, Phases of a Compiler and Bootstrapping. (7L)
Understand Lexical Analyzer, Transition Diagram of different tokens, Reserved Word Strategy. (7L)
Idea of Syntax Analyzer, Ambiguity, Parse Tree, Top Down and Bottom Up Parser. (8L)
Concept of Semantic Analyzer, Semantic Actions, Intermediate Code, Virtual Machine. Lexical and Grammatical Errors. (7L)
Idea of Code Optimization, Criterion of Optimization, Different Local and Global Optimization Techniques. (7L)
Idea of Code Generation, Instruction Costs, Code Generation Algorithm, Run Time Store Management. (8L)
Symbol Table Design, Fixed Length and Variable Length Entry, Symbol Table Actions, Different Searches, Hash Table Organization, Different Deletions of Symbols, Linked List and Tree Representation. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Principles of Compiler Design – Alfred V. Aho & Jefrey D. Ullman, Pearson Education.

Reference Books:
1.Compiler Design in C – Holub, Prentice Hall.</td></tr>
</table>

CSC 504 Microcontroller based Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 504</td><td>Microcontroller based Systems</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Remember the architecture and instruction sets of PIC and ARM.
● CO2: Understand PIC interrupts, interfacing of peripherals.
● CO3: Apply the knowledge in LCD keyboard interfacing, ADC, DAC and Sensor interfacing and ARM assembly language programming.
● CO4: Analyze ADC, DAC and Sensor interfacing using PIC; relate PIC and ARM architectures.
● CO5: Appraise the architecture of PIC and ARM in terms of RISC architecture.
● CO6: Create embedded ARM applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT I INTRODUCTION TO PIC MICROCONTROLLER 9 14
Introduction to PIC Microcontroller–PIC 16C6x and PIC16C7x Architecture–PIC16cxx–-Pipelining - Program Memory considerations – Register File Structure - Instruction Set - Addressing modes –Simple Operations. (12L)

UNIT II INTERRUPTS AND TIMER 9 PIC Microcontroller Interrupts- External Interrupts-Interrupt Programming–Loop time subroutine - TimersTimer Programming–Front panel I/O-Soft Keys– State machines and key switches– Display of Constant and Variable strings. (8L)

UNIT III PERIPHERALS AND INTERFACING 9 I 2 C Bus for Peripherals Chip Access– Bus operation-Bus subroutines– Serial EEPROM—Analog to Digital Converter–UART-Baud rate selection–Data handling circuit–Initialization - LCD and keyboard Interfacing - ADC, DAC, and Sensor Interfacing. (8L)

UNIT IV INTRODUCTION TO ARM PROCESSOR 9 ARM Architecture –ARM programmer’s model –ARM Development tools- Memory Hierarchy–ARM Assembly Language Programming–Simple Examples–Architectural Support for Operating systems. (10L)
UNIT V ARM ORGANIZATION 9 3-Stage Pipeline ARM Organization– 5-Stage Pipeline ARM Organization–ARM Instruction ExecutionARM Implementation– ARM Instruction Set– ARM coprocessor interface– Architectural support for High Level Languages – Embedded ARM Applications. (4L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7"></td></tr>
</table>

	Text Books: 1. Peatman,J.B., “Design with PIC Micro Controllers”PearsonEducation,3rdEdition, 2004. 2. Furber,S., “ARM System on Chip Architecture” Addison Wesley trade Computer Publication, 2000. **Reference Books:** 1. Mazidi, M.A.,“PIC Microcontroller” Rollin Mckinlay, Danny causey Printice Hall of India, 200

CSS 551 Design and Analysis of Algorithms Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 551</td><td>Design and Analysis of Algorithms Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2">Design and analysis of Algorithm (CSC 503), Data Structures and Algorithms Laboratory (CSS 352)</td><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Can identify the essence of theory into implementation.
● CO2: Able to interpret the theory efficiently through coding.
● CO3: Able to verify the theory experimentally.
● CO4: Can explain the behaviour of an algorithm efficiently.
● CO5: Can compare the efficiency of different algorithms.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">Assignment 1: Exponential versus Polynomial Running time solution of a problem.
Assignment 2: Heaps and priority queue.
Assignment 3: Problem based on Linear time sorting algorithm.
Assignment 4: Problem using Divide and Conquer algorithm.
Assignment 5: Problem using Greedy algorithm.
Assignment 6: Problem using Dynamic Programming algorithm.
Assignment 7: Graph representation and traversal.
Assignment 8: Problem using Union Find structure.
Assignment 9: Problem using Interval tree.
Assignment 10: Convex Hull computation from a given set of n points in 2D and then determining the farthest pair of these point set.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, Introduction to Algorithms, by Prentice Hall India.
2. J. Kleinberg and Eva Tardo, Algorithm Design by Pearson Education (Indian edition).

Reference Books:

1. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.
2. S. Dasgupta, C. Papadimitriou and U. Vazirani, Algorithms, by Tata McGraw-Hill.

Others:
The Algorithm Design Manual 2nd ed. 2008 Edition by Steven S S. Skiena, Springer.</td></tr>
</table>

CSS 552 Microcontroller based System Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 522</td><td>Microcontroller based Systems Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Remember basics of Audruino and Sensor interfacing.
● CO2: Understand Audrino based applications using sensors.
● CO3: Apply gathered knowledge in design of Audruino based systems.
● CO4: Analyze design of Audriono based systems.
● CO5: Defend the design of the system using Aurduino.
● CO6: Create new embedded Aurduino based systems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Assignment 1: Familiarization with Aurdino.
Assignment 2: LED with Aurdino.
Assignment 3: ADC/DAC interfacing with Aurdino.
Assignment 4: Keyboard interfacing with Aurdino.
Assignment 5: Traffic controller using Aurdino.
Assignment 6: Display interface with Aurdino.
Assignment 7: Robot using Aurdino.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Peatman, J.B., “Design with PIC Micro Controllers” Pearson Education, 3rd Edition, 2004.
2. Programming Arduino: Getting Started with Sketches Book by Simon Monk.</td></tr>
</table>

CSS 553 Operating Systems Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 553</td><td>Operating Systems Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">Introduction to Computing (CSC01), Data Structures and Algorithms (CSC303)</td><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Implement elementary UNIX system commands.
● CO2: Devise programs to test synchronization problems.
● CO3: Design and develop user level thread library.
● CO4: Design and implement file system.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Assignment 1: Getting a feel of race conditions through read/write operations by multiple process (run the same program in four terminals simultaneously) on a single binary file.
Assignment 2: Design application where the parent process uses fork system call to create multiple child processes in the different given hierarchy and displaying and storing the process hierarchy in a separate file.
Assignment 3: Design application where parent sync with several child processes using fork & wait system call to solve a particular task (searching, prime number generation, etc.) like parallely also try to understand and change process priorities using system calls.
Assignment 4: Implement signal handling among parent child processes.
Assignment 5: Design multithreaded application using POSIX thread library.
Assignment 6: Create shared memory to be used among a set of concurrent processes using POSIX library.
Assignment 7: Implement semaphores (named) and solve data access sync problems like (producer/consumer) using multiple processes.
Assignment 8: Implement semaphores (unnamed) and solve data access sync problems like (producer/consumer) using multiple threads.
Assignment 9: Use other IPC mechanisms like message queues, named pipe.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books: “Beginning Linux Programming”, 4th Edition by Richard Stones, Neil Matthew, Wiley Publishing, Inc.

Reference Books: “Advanced Programming in the UNIX environment”, 3rd Edition, W. Richard Stevens and Stephen A. Rago, Addison-Wesley, 2013.</td></tr>
</table>

Open Elective – 2

CSO 541 Fundamentals of Algorithms 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO541</td><td>Fundamentals of Algorithms</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Data</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Will be able to analyse the time complexity of algorithms.
● CO2: Able to map real life problems into algorithmic framework.
● CO3: Will have concept of different algorithm design paradigm.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Non-linear data structures. Trees. Binary search trees, AVL tree. (5L)
2. Set Representations. Disjoint Set Union. Priority Queues. (4L)
3. Graph Representations. AND-OR graphs. BFS. DFS. (4L)
4. Algorithm analysis techniques, asymptotic complexity, Big-Oh, Big-omega and Theta notation, Lower bound analysis. (5L)
5. Divide and Conquer. Analysis of Binary Sort, Merge sort, Heap sort, Quicksort, Selection problem, Multiplication of two large n-bit numbers, Strassens' Matrix Multiplication. (7L)
6. Greedy Techniques. Minimal Spanning Trees, Knapsack problem, Huffman's Codes. Job Scheduling. (6L)
7. Dynamic Programming. All Pairs. Shortest Paths, Matrix Chain Multiplication Problem, Traveling Salesperson Problem. (5L)
8. Backtracking. N-Queens problem. Sum of Subsets. (3L)
9. Introduction to NP Hard problems. (3L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, Introduction to Algorithms, by Prentice Hall India.
2. J. Kleinberg and Eva Tardo, Algorithm Design by Pearson Education (Indian edition).

Reference Books:
1. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.</td></tr>
</table>

CSO542 Database Management System 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO542</td><td>Database Management System</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Fundamental knowledge in Programming and Data Structures</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the basic concepts and appreciate the applications of database systems.
● CO2: Comprehend the fundamentals of design principles for logical design of relational databases.
● CO3: Apply the query writing skill and its subsequent optimization.
● CO4: Discuss the basic issues of transaction processing and concurrency control.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Concept & Overview of DBMS, Applications, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS. (3L)

Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram. (5L)

Relational Model: Structure of relational Databases, Various Relational Algebra operations used to write a query, Views. (5L)

SQL: Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Referential views, Nested Subqueries. (5L)

Index Structures: Necessity of index structures, Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes. (3L)

Normalization: Functional Dependency, Anomalies in a Database, The normalization process: Conversion to first normal form, Conversion to second normal form, Conversion to third normal form and BCNF, Fourth Normal form and fifth normal form, Denormalization, Loss-less join decomposition, Dependency preservation. (6L)

Transaction processing: Introduction of transaction processing, advantages and disadvantages of transaction processing system, online transaction processing system, serializability. (4L)

Concurrency Control: Serializability by Locks, Lock Modes, Lock based Concurrency Control, Concurrency Control by Timestamps. (4L)</td></tr>
</table>

	Query Optimization: Heuristics in Query Optimization, Converting Query Tree to Query Evaluation Plan. (3L) **Distributed Database (DDB):** Introduction of DDB, DDBMS architectures, Data Replication, Data Fragmentation. (4L)
Text Books, and/or reference material	**Text Books:** 1. "DatabaseSystem Concepts", Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw-Hill. 2. "Distributed Databases Principles & Systems", Stefano Ceri and Giuseppe Pelagatti, McGraw-Hill International Editions. **Reference Books:** 1. "Fundamentals of Database Systems", Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley.

CSO543 Computer Organization 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO543</td><td>Computer Organization</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">CSC01 (Introduction to Computing)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Analyze the various parts of a modern computer functional units, bus structure, addressing modes and Computer arithmetic.
● CO2: Identify the process involved in executing an instruction and fetching the word from memory.
● CO3: Design the hardwired and micro-programmed control units and implementation of interrupts.
● CO4: Understand the memory hierarchy and design a memory system.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT-I: Introduction: Evolution of computers, Basic Structure of Computers: Basic Operational Concepts, GPR based and stack based organisation. Bus Structures, Performance Measurement: Processor Clock, Basic Performance Equation, Clock Rate, Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes. (10L)

UNIT-II: Fundamental concepts of the processing Unit: Fetching and Storing words, Register Transfer, Execution of instruction, Arithmetic Operations: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Combinational and Sequential ALU, ALU expansion strategies, Floating Point Numbers (IEEE754), Floating Point Operations. (10L)

UNIT-III: Computer Organization and Design (Datapath and control path): Instruction codes, computer registers, computer instructions, timing & control, instruction cycle, memory reference instructions, Hard-wired Control, Micro programmed Control: Micro instruction, Microprogram sequencing, Input/output Organization: Accessing I/O Devices, Interrupt, Bus Arbitration schemes. (Brief overview of 8085/8086 microprocessor). (12L)

UNIT-IV: Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, page mode access, interleaved access. Performance Considerations, Virtual Memories, Secondary Storage. (10L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. David A Patterson, John L Hennessy, “Computer Organization and Design”, (The Hardware/Software Interface) Morgan Kaufmann.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill.

Reference Books:
1. William Stallings, “Computer Organization and Architecture”.
2. Nicholas P Carter, “Computer Architecture & Organisation”.</td></tr>
</table>

CSO544 Operating Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 544</td><td>Operating Systems</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Introduction to Computing (CSC01), Data Structures and Algorithms (CSC303)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding functional architecture of an operating system.
● CO2: Understanding process control algorithms, deadlocks and multi-threading in OS.
● CO3: Develop application programs using UNIX system calls.
● CO4: Design and solve control & data access synchronization problems.
● CO5: Understanding virtual memory organization and management in OS.
● CO6: Understand standard FAT & UNIX file system.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introductory Concepts: Introduction to Operating System as a whole, memory, CPU(registers and ALU), Evolution of Operating System-types of OS(advantages and drawbacks), Performance measurement metrics. (5L)

Process Data Structures and State transitions: Process management, Basic Definitions, Process table, PCB(process control block), PTE(process table entry), Process states, Transition diagram, context of process-user level, kernel-level and process Level. (4L)

Process Control: Process creation, Parent and Child processes, System calls--fork(), exit(), wait(), kill(), Signal handling, Process scheduling strategies-FCFS, SPN, SRT, Round Robin, HRRN, Fair share scheduling. (6L)

Multi-threading: Threads in OS, thread vs process, ULT & KLT, Applications of threads, Use of POSIX threads library. (4L)

Process synchronization - Race condition, Critical section, Process Sync Solution using Algorithmic approach (Lamport bakery Algorithm), Creating shared memory using POSIX library. (3L)

Semaphore- Binary and Counting semaphore, P() and V() operations, Solving Classical problem using semaphores- Sleeping barber, Producer-consumer, Reader-writer, Dining philosophers’s problem, Posix library for semaphores. (7L)

Monitors - Solving Classical problems using monitors. (4L)

Deadlocks - Necessary and sufficient conditions for deadlocks, approaches to deal with deadlocks, Deadlock Prevention, Avoidance (Banker’s algorithm) and Detection. (3L)</td></tr>
</table>

	Memory organization & management - Virtual memory organization, Pure Paging, Pure Segmentation, Combined Paging-Segmentation, Inverted PMT, Page fault handling algorithms, Working set theory. (7L) **File management**- Directory structure, Storage of files on disks, contiguous and non-contiguous file allocation strategies, Internal and external fragmentation, FAT & Inode Structure, Free Space management, Disk scheduling strategies. (6L) **I/O management concepts** (2L)
Text Books, and/or reference material	**Text Books:** 1. "Operating System Concepts", Silberschatz and Galvin. 2. "Operating Systems: Internals and Design Principles" by William Stalling. 3. "Operating Systems: A Concept-Based Approach" by D M Dhamdhere. **Reference Books:** 1. "Operating System: A Design-oriented Approach" by Charles Crowley. 2. "Operating Systems: A Modern Perspective" by Gary J Nutt. 3. "Design of the Unix Operating Systems" by Maurice Bach. 4. "MODERN OPERATING SYSTEMS" by Andrew S Tanenbaum. **Others:** ● https://nptel.ac.in/courses/106/106/106106144/# Course "Introduction to Operating Systems" by PROF. CHESTER REBERIO, IIT Madras. ● https://nptel.ac.in/courses/106105214/ Course "Operating System Fundamentals" by Prof. Santunu Chattopadhyay, IIT Kharagpur.

SIXTH SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	HSC631	Economics and Management Accountancy	3-0-0	3	3
2	CSC601	Software Engineering	3-0-0	3	3
3	CSC602	Data Communication and Computer Networks	3-1-0	4	4
4	CSE610 --	Depth Elective - 1	3-0-0	3	3
5	CSE610 --	Depth Elective - 2	3-0-0	3	3
6	CSS651	Compiler Laboratory	0-0-3	1.5	3
7	CSS652	Data Communication and Computer Networks Laboratory	0-0-3	1.5	3
8	CSS653	Database Management System Laboratory	0-0-3	1.5	3
9	XXS681	Co-curricular Activities - VI (Optional)	0-0-0	0	0
		TOTAL	**15-1-9**	**20.5**	**25**

6th Semester

	Basket of Depth Elective – 1, 2
CSE611	**Embedded System Design**
CSE612	**System Software**
CSE613	**Internet and Web Technologies**
CSE614	**Advanced Computer Architecture**
CSE615	**Optimization Techniques**
CSE616	**Artificial Intelligence**
CSE617	**Advanced Algorithms**
CSE618	**Information Coding Theory**
CSE619	**Computer Graphics**
CSE620	**Game Theory and its Applications**
CSE621	**Digital Systems Testing**
CSE622	**Soft Computing**
CSE623	**Advanced Database Systems**

CSC 601 Software Engineering 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 601</td><td>Software Engineering</td><td>PCR</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment.
● CO2: An ability to work in one or more significant application domains.
● CO3: Work as an individual and as part of a multidisciplinary team to develop and deliver quality software.
● CO4: Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle.
● CO5: Demonstrate an ability to use the techniques and tools necessary for engineering practice.
● CO6: To manage time, processes and resources effectively by prioritizing competing demands to achieve personal and team goals Identify and analyses the common threats in each.</td></tr>
<tr><td>Topics Covered</td><td colspan="7"><u>Software Paradigm / Introduction</u>: Definition of Information System, software, software engineering paradigms, Software engineering in context of Business Process Engineering, Goal of Software Engineering, Quality focus. (2L)

<u>Software Process Model</u>: Umbrella activities; Waterfall Model, Prototype model, Rapid Application Development Model, Evolutionary Approach in Process model (Spiral Model). (4L)

<u>Requirement Engineering</u>: Requirements Engineering Tasks, Information Modelling (Entity Relationship Model, Extended ER Model), Functional Model (DFD, CFD), Behavioral Model (State Transition Diagram), Petri-net modelling, System Requirement Specification (SRS), Specification Language – Formal Methods, Regular Expression, Decision Tree, Decision Table, SRS Standards. (6L)

<u>Design Principle and Basics</u>: Design level tasks, Problem partitioning, abstraction, top down & bottom up design strategies, refinement techniques, Minor Design principles, Control Hierarchy (Structured Chart), constraint design (Warnier –Orr). (2L)

<u>Design Language basics</u>: Unified Modelling Language – Building Blocks, Well-formedness rule; Use case, structural diagram introduction - Class Diagram, Object Diagram, Sequence diagram, collaboration diagram. (6L)

<u>Modular Design</u>: Concept of module and Modular design, Functional independency, Cohesion, Coupling, measuring cohesion and coupling, Model Driven Architecture. (4L)</td></tr>
</table>

	Architecture Basic: Software architecture, Functional and extra-functional properties, families of related system, Architectural styles: Data-centric, data-flow, call and Return, layered, enterprise. (2L) *Project Management*: LOClFunction Point Analysis PERT Chart estimation, Different cost estimation: Delphi-empirical-COCOMO estimation. (2L) *Coding Techniques & Standard guidelines*: Rules/guidelines for standard Coding l Gunning Fog Index for documentation. (2L) *Testing strategy 1*– Introduction to Software Testing, Software Testing Terminology and Methodology Verification and Validation, Static Testing: Inspections, Structured Walkthroughs, Technical Reviews l Dynamic Testing: Black-Box Testing Techniques: Boundary Value Analysis (BVA), Equivalence Class Testing, State Table-Based Testing, Decision Table-Based Testing, Cause-Effect Graphing Based Testing, Error Guessing Dynamic Testing : White-Box Testing Techniques: Need of White-Box Testing, Logic coverage Criteria, Basis Path Testing, Graph Matrices, Loop Testing, Data Flow Testing.(6L) *Testing strategy 2*- Validation Activities: Unit Validation Testing, Integration Testing, Function Testing, System Testing, Acceptance Testing lRegression Testing: Progressive vs Regressive Testing, Regression Testability. (2L) *Software & Metrics:* Software Measurement & metrics, Direct and indirect metrics, Size oriented metrics, Function oriented Metrics, Complexity Metrics – McCabe Complexity, McClure Complexity, and Halstead Software Science. (4L)
Text Books, and/or reference material	**Text Books:** R. S. Pressman -"Software Engineering – Practitioner's Approach"- McGraw Hill International. I. Somerville – "Software Engineering", Addison-Wesley **Reference Books:** Rajib Mal - "Fundamental of Software Engineering", PHI. **Others:** Unified Modelling Language, Object Management Group, http://www.omg.org/spec/UML/

CSC 602 Data Communication and Computer Networks 3-1-0 4Credits 4Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSC 602</td><td>Data Communication and Computer Networks</td><td>PCR</td><td>3</td><td>1</td><td>0</td><td>4</td><td>4</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Data Structures and Algorithms, Operating system concepts</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
● CO2: Comprehend the fundamentals of Physical layer, and will apply them in real time applications.
● CO3: Identify data link layer concepts, design issues, and protocols.
● CO4: Classify the routing protocols and analyze how to assign the IP addresses for the given network.
● CO5: Acquire knowledge of Application layer and Presentation layer paradigms and protocols.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Overview of Data Communication and Networking: Introduction; Data communications: components, data representation (ASCII,ISO etc.), direction of data flow (simplex, half duplex and full duplex); network criteria, physical structure (type of connection, topology), categories of network (LAN, MAN,WAN); Internet: brief history, Protocols and standards; Reference models: OSI reference model, TCP/IP reference model, their comparative study. (4L)

Physical Level: Overview of data (analog & digital), signal (analog & digital), transmission (analog & digital) & transmission media (guided & unguided); Circuit switching: time division & space division switch, TDM bus; Telephone Network. (6L)

Data link Layer: Types of errors, framing (character and bit stuffing), error detection & correction methods; Flow control; Protocols: Stop & wait ARQ, Go-Back- N ARQ, Selective repeat ARQ, HDLC; Medium Access sublayer: Point to Point Protocol, LCP, NCP, Token Ring; Reservation, Polling, Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA, Traditional Ethernet, Fast Ethernet. (12L)

Network layer: Internetworking & devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway; Addressing: IP addressing, subnetting; Routing: techniques, static vs. dynamic routing, Unicast Routing Protocols: RIP, OSPF, BGP; Other Protocols: ARP, IP, ICMP, IPV6, Congestion Control: Open Loop, Closed Loop choke packets; Quality of service: techniques to improve QoS: Leaky bucket algorithm, Token bucket algorithm. (14L)

Transport layer: Process to Process delivery; Socket address, UDP; TCP. (4L)

Application Layer: Introduction to DNS, SMTP, SNMP, FTP, HTTP & WWW. (4L)</td></tr>
</table>

	Security: Cryptography (Public, Private Key based), Digital Signature, Firewalls. (4L) **Modern topics:** ATM, DSL technology, Introduction to blue-tooth. (4L) **Queuing Theory:** Introduction to Queuing Theory and Delay Analysis for networks. (4L)
Text Books, and/or reference material	**Text Books:** 1.B. A. Forouzan – "Data Communications and Networking (3rd Ed.) " – TMH. 2. A. S. Tanenbaum – "Computer Networks (4th Ed.)" – Pearson Education/PHI. **Reference Books:** 3. Comer – "Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)" – Pearson Education/PHI.

CSS 651 Compiler Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS651</td><td>Compiler laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">Compiler Design
Theory of Computation CSC402</td><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To apply the concept of regular expressions in the identification of tokens in a lexical analyzer.
● CO2: To explore the use of program generating softwares like LEX and FLEX.
● CO3: To generate context -free grammar to represent the syntax of the language.
● CO4: To use compiler generators like YACC and BISON.
● CO5: To use syntax directed translation to generate intermediate code.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Handle tokens in an input using LEX generated program.
2. Describe class of tokens using regular expressions in LEX.
3. Use context free grammars with YACC to describe simple syntactic structures.
4. Remove ambiguity in if-then-else constructs using YACC’s inbuilt features.
5. Use syntax directed translation in YACC to generate simple intermediate code.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Lex - A Lexical Analyzer Generator M. E. Lesk and E. Schmidt Online Manual.
2. Yacc: Yet Another Compiler-Compiler Stephen C. Johnson Online Manual.
3. Lex & Yacc John R. Levine, Tony Mason, Doug Brown , O'Reilly & Associates.
Reference Books:
1. Compilers: Principles, Techniques, and Tools
By Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman. Addison-Wesley Pub Co.</td></tr>
</table>

CSS 652 Data Communication and Computer Networks Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 652</td><td>Data Communication and Computer Networks Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2">Operating System Laboratory</td><td colspan="6">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Develop programs for client-server applications.
● CO2: Perform packet sniffing and analyze packets in network traffic.
● CO3: Implement error detecting and correcting codes.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Assignment 1 : Packet capturing and analyzing using wireshark packet sniffer tool
Assignment 2 : Socket Programming for TCP client server (Iterative server).
Assignment 3 : Socket Programming for TCP client server (Concurrent Server).
Assignment 4 : Socket programming for UDP client.
Assignment 5 : Handling both TCP client and UDP client using select() system call.
Assignment 6 : Simplified FTP implementation.
Assignment 7 : Two player game (Tic Tac Toe) implementation.
Assignment 8 : Implementation of CRC and Hamming code for error handling
Assignment 9 : RPC (Remote Procedure Call) implementation.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Richard Stevens, Unix Network Programming, Volume 1 and 2, Addison-Wesley Professional.
Reference Books:
1. Neil matthew and Richard Stones, Beginning Linux Programming, Wrox Publishers, 4th Edition.</td></tr>
</table>

CSS 653 Database Management System Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 653</td><td>Database Management System Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="4">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="3">Programming knowledge, Data structure knowledge</td><td colspan="4">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Understand, appreciate and effectively explain the underlying concepts of database technologies.
● CO2: Design and implement a database schema for a given problem-domain.
● CO3: Populate and query a database using SQL DML/DDL commands.
● CO4: Programming PL/SQL including stored procedures, stored functions, cursors, packages.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">Structured Query Language (SQL):
1. Creating Database Creating a Database Creating a Table Specifying Relational Data Types Specifying Constraints Creating Indexes.

2. Table and Record Handling INSERT statement Using SELECT and INSERT together DELETE, UPDATE, TRUNCATE statements DROP, ALTER statements.

3. Retrieving Data from a Database The SELECT statement Using the WHERE clause Using Logical Operators in the WHERE clause Using IN, BETWEEN, LIKE, ORDER BY, GROUP BY and HAVING Clause Using Aggregate Functions Combining Tables Using JOINS Subqueries.

4. Database Management Creating Views Creating Column Aliases Creating Database Users Using GRANT and REVOKE.

PL / SQL:
Decision-control in PL / SQL, Cursors in PL / SQL, Stored Procedures.

Case Studies: Real-life case studies.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books: SQL, PL/SQL the Programming Language of Oracle by Ivan Bayross, PHI, 2010.
Reference Books: SQL The Complete Reference, Groff James, 3rd Edition, Tata McGraw-Hill Education, India.</td></tr>
</table>

Depth Elective – 1, 2

CSE 611 Embedded System Design 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 611</td><td>Embedded System Design</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Computer Organization and Architecture</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the concept of embedded system and the architecture of such system.
● CO2: Understand the role of controller, timer and interfaces for embedded system.
● CO3: Design and analyzes the various scheduling algorithm and protocols for power efficient embedded system.
● CO4: Understand the concept of HW-SW partition and codesign principles.
● CO5: Understand the modeling and specification of embedded system.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT-1: Introduction to embedded system, Challenges in Embedded System Design, Processors: General Purpose and ASIPs Processor, Instruction Set Architecture: CISC and RISC instruction set architecture, Basic Embedded Processor/Microcontroller Architecture, DSP Processors, PIC, designing a Single Purpose Processor, Optimization Issues, Introduction to FPGA, Behavior Synthesis on FPGA using VHDL. (8L)

UNIT-II: Sensors and Signals, Discretization of Signals and A/D Converter, Quantization Noise, SNR and D/A Converter, Arduino Uno, I/O Devices, Timers and Counters, Watchdog Timers, Interrupt Controllers, Serial Communication, Controller Design using Arduino. (8L)

UNIT-III: Power Aware Embedded System, SD and DD Algorithm, Parallel Operations and VLIW, Code Efficiency, DSP Application and Address Generation Unit. (8L)

UNIT-IV: Real Time OS, RMS Algorithm, EDF Algorithm and Resource Constraint Issue, Priority Inversion and Priority Inheritance Protocol. (7L)

UNIT-V: Modeling and Specification, FSM, State chart and Statemate Semantics, Program State Machines, SDL, Data Flow Model, Hardware Synthesis, Scheduling, Case study: Digital camera design. (6L)

UNIT-VI: HW-SW Partitioning, Optimization, Simulation, Formal Verification. (5L)</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. Frank Vahid, Tony Givargis "Embedded System Design: A Unified Hardware / Software Introduction". 2. - D.D. Gajski, S. Abdi, A. Gerstlauer, G. Schirner, "Embedded System Design: Modeling, Synthesis and Verification" **Reference Books:** 1.Peter Marwedel, "Embedded System Design".

CSE 612 System Software 3-0-0 3 Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE612</td><td>System Software</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites: Programming Language Paradigms, Theory of Computing, Computer Architecture, Operating Systems, Compilers</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To introduce the students to the collection of programs and procedures which constitute the system software of a computer platform.
● CO2: To allow the students to understand & acknowledge the main objectives, problems faced and programming techniques used by a system programmer in designing and implementing system software.
● CO3: To emphasize on conceptual framework in which the system software is developed and used rather than a broad overview of programs which belong to the system software running on a particular computer platform.
● CO4: To enable the students to deduce the logical relationship between the software components of any software system.
● CO5: To enable students to understand the mechanism of Integration of different System Software components.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Part I: The Methodology
● The fundamental objective of this part is to develop a concept of a System.
○ Concept is to be built upon both Mathematical construction(Algebraic and Logic Systems) as well as around construction based on Abstract Machines. (3L)
● Programs and documents that are part of System Software are to be defined. (1L)
● A structuring of System Software Components are to be defined and built.
○ Vertical Structuring: Components of the system software are layered on a hierarchy of levels. The hardware system is taken as the first level of this hierarchy. The interface relationship between the components of the system software vertical hierarchy is then established. (2L)
○ Horizontal Structuring: Each level of the system software vertical hierarchy is discussed as a horizontal structure. The elements of this horizontal structure (formal definitions will also be given) are specific software components of the system software organized as software systems. (2L)
● The specific problems posed by the interaction between the software system components of a horizontal level of the system software hierarchy are discussed and illustrated. The problems raised by the reliability, efficiency, convenience, and evolution of a system software are introduced and illustrated. (2L)

Part II: Programming Support Environment:</td></tr>
</table>

	● of a System Software is to be discussed as the collection of tools offered by a computer platform to computer users to help them use the computer to develop programs that solve their problems. (2L) ● Detailed discussions on Topics like: Language, Translators, Interpreters, Mechanism of target machine code generation; proper emphasis on distinguishing between Compilers, Assemblers, Linker/Loaders, and Interpreters will be there. Interfacing users with the Operating System environment as tolls from the support environment is to be discussed. (6L) ● Case study of JVM, GNU GCC implementation of the Linux Assembler, Linker and Loader will be dealt with in detail, introducing implementation of symbol tables. (8L) Part III: Execution Support Environment: ● A software system that manages computer resources of the computer platform and the processes running on the computer platform will be introduced and illustrated by the operating system. (3L) ● The components of the operating system itself are layered on the levels of a hierarchy. (2L) ● The mechanism of a system call (system function call) will be discussed as a tool for implementing this hierarchy relation. (2L) ● The following layers of an operating system will be discussed with a practical illustration with the Linux kernel, with mechanisms of designing system programs developed with and for the support of: (10L) ○ Interrupt System ⇔ designing interrupt handlers. ○ Process Management System ⇔ designing schedulers. ○ Memory Management System ⇔ designing page-fault exception handlers ○ Input/Output Management System ⇔ designing device drivers ○ Information Management System (File System) ⇔ examining ext2/ext3/ext4.
Text Books, and/or reference material	**Text Books:** 1. System Software and Software Systems: Systems Methodology for Software, Tudor Rus, World Scientific Press, 1993. 2. System Software: An Introduction to Systems Programming, leyland L. Beck, 1996. 3. System Programming with C and Unix, Adam Hoover, Adison Wesley 2010. **Reference Books:** 1. Understanding the Linux Kernel, Daniel P. Bovet, Marco Cesati, O'Reilly Pub Date:November 2005. Available online at: http://johnchukwuma.com/training/UnderstandingTheLinuxKernel3rdEdition.pdf

CSE 613 Internet and Web Technologies 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 613</td><td>Internet and Web Technology</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Programming Fundamentals, Data Structure and Algorithms, Operating Systems, Data networks (may be carried out simultaneously)</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding the fundamental concepts of Internet Structure and Protocols.
● CO2: Using TCP/IP protocols and Internet programming using SOCKET API.
● CO3: Understanding HTTP protocol and Structures of Web Programming.
● CO4: Designing and developing Web applications with security enhancement.
● CO5: Understanding Semantic Web and Applying Web Analytics over Semantic Web.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">INTERNET TECHNOLOGY:
Brief review of Data Networking; data transmission, links and MACs, Forwarding and Routing, TCP-IP layered network concepts.
(3L)
Internet specific issues like scalability, inter-operability.
(1L)
Internet Structures – logical and physical grouping with sub-netting and super netting.
(3L)
Review of TCP-IP protocols – processing, performance and variations.
(3L)
Security Implementations - secured IP, Transport Layer security.
(3L)
Quality of Service Issues and their Application in Internet.
(2L)

SOCKET PROGRAMMING: Introduction to SOCKET API; Client programming; Server programming – sequential, concurrent and multi-threaded; P2P application Programming.
(4L)

HTTP: Requests and Responses - Message Formats, Headers and Fields; TCP Keep-alive and pipe-lining concepts; Server Architecture ,Performance and Deployment.
(3L)

WEB PROGRAMMING: Document Object Model; Client side scripting fundamentals: Server Side Scripting and Programming – Data base connectivity, session management and security enhancement; Introduction to Web Application Development Platforms – JavaEE, Dzango.
(7L)</td></tr>
</table>

	XML: DTD and Schema; Visualisation using XSLT; Web Application using XML; Service Oriented Architecture and Web services based application development and deployment; Xquery and SOA based application development platforms. (6L) SEMANTIC WEB: General Concept of Semantic Web and linked Data; RDF based relation description; Web Ontology concepts and use; Putting XML, RDF and Ontology together to develop semantic web applications; Capturing Information from semantic web pages; Data analytics over semantic and linked Web. (7L)
Text Books, and/or reference material	**Text Books:** 1. B. A. Forouzan, "TCP/IP Protocol Suite", 4th Edition, 2010, McGrawHIll Publishers. 2. P. Deitel, H. Deitel, A Deitel, "Internet and World Wide Web – How to Program", Pearson. 3. G. Antoniou, P. Groth, F. Harmelen and R. Hoekstra, "A Semantic Web Primer" Prentice Hall India. **Reference Books:** 1. D. E. Comer and D L Stevens, "Internetworking with TCP/IP vol.II", Pearson.

CSE 614 Advanced Computer Architecture 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 614</td><td>Advanced Computer Architecture</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Digital Electronics, Computer Organisation</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To know about the classes of computers, and new trends and developments in computer architecture.
● CO2: To acquire knowledge about the various architectural concepts that may be applied to optimize and enhance the classical Von Neumann architecture into high performance computing systems.
● CO3: To learn the basic design procedure for different levels of parallelism.
● CO4: To learn the design issues relating to the architectural options.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">OVERVIEW OF VON NEUMANN ARCHITECTURE: Instruction set architecture; The Arithmetic and Logic Unit, The Control Unit, Memory and I/O devices and their interfacing to the CPU; Measuring and reporting performance; CISC and RISC processors. (4L)

PIPELINING: Pipelining fundamentals, Linear and Nonlinear Pipeline Processors, Arithmetic and instruction pipelining, Pipeline hazards, Techniques for overcoming or reducing the effects of various hazards, superscalar and super pipelined and VLIW architectures. (8L)

INSTRUCTION –LEVEL PARALLELISM (ILP): Concepts and challenges of ILP; Compiler Techniques for exposing ILP; Branch costs reductions - Static and Dynamic predictions; Hardware-based speculation. (8L)

MULTIPROCESSORS ARCHITECTURES: Introduction; Taxonomy of parallel architectures, Centralized shared-memory architecture: synchronization, memory consistency, interconnection networks. Distributed shared-memory architecture. (8L)

MEMORY HIERARCHY DESIGN: Introduction; Memory technology and optimizations, Virtual memory, Cache memory, Cache performance; Cache Optimizations, Cache coherence, Cache coherence protocols – snoop based and directory based protocols, Advanced optimizations of cache performance. (10L)

INTERCONNECTION NETWORKS: Topology, Different interconnection Networks, Routing Mechanism. (4L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

1. Computer Architecture, A Quantitative Approach – John L. Hennessey and David A. Patterson; 4th edition, Morgan Kaufmann.</td></tr>
</table>

	2. Advanced Computer Architecture Parallelism, Scalability, Programmability – Kai Hwang; Tata Mc-Graw Hill. **Reference Books:** 1. Computer architecture and parallel processing – Kai Hwang and FayéAlayé Briggs; McGraw-Hill. 2. Parallel Computer Architecture, a Hardware / Software Approach – David E. Culler, Jaswinder Pal Singh, Anoop Gupta; Morgan Kaufman. 3. John Paul Shen and Mikko H. Lipasti, Modern Processor Design: Fundamentals of Superscalar Processors, Tata McGraw-Hill. 4. M. J. Flynn, Computer Architecture: Pipelined and Parallel Processor Design, Narosa Publishing House. **Others:** NPTEL/MOOC Courses materials.

CSE 615 Optimization Techniques 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE615</td><td>Optimization Techniques</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Engineering Mathematics, Discrete Mathematics</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To understand the Basic principles of optimization.
● CO2: To able to formulate optimization problem mathematically.
● CO3: To know various solution methods in optimization Problems.
● CO4: Able to perform sensitivity analysis and post processing of optimal solutions.
● CO5: Able to explore a wide range of engineering optimization problems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Optimization- Development, mathematical problem formulation, engineering applications of optimization, classification of optimization problems. (3L)

Classical Optimization of Single and Multi variable- Optimality criterion for single and multi-variable method, Region elimination methods, Gradient based methods for single variable and Multivariable, unidirectional search, direct search methods. (10L)

Linear Programming- Standard form of linear programming (LP) problem, Graphical method, Simplex algorithm, Simplex criterion, Duality in LP, Sensitivity or post optimality analysis, Transportation Problem and Assignment Problem. (12L)

Dynamic Programming- Introduction, Sequential optimization, computational procedure, discrete versus continuous dynamic programming, curse of dimensionality. (3L)

Integer Programming- Introduction, Linear and Nonlinear integer programming, Methods for integer programming. (2L)

Non-Linear Programming- Introduction, examples of non-linear programming, types of non-linear programming, Constraint and Unconstrained optimization, methods of nonlinear programming. (7L)

Modern Optimization- Multi-objective optimization, many optimization, Genetic Algorithms, Particle Swarm Optimization, Differential Evolution, CMA-ES, applications in engineering optimization problems. (5L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. S. S. Rao, Engineering Optimization: Theory and Practice, New Age International.
2. K. Deb, Optimization for Engineering Design, Prentice Hall of India.
3. A. Ravindran, K. M. Ragsdell and G. V. Reklaitis, Engineering Optimization: Methods and Applications, Wiley.
4. Hillier & Lieberman, Introduction to Operations Research, TMH.</td></tr>
</table>

	Reference Books: 1. S. M. Sinha, Mathematical Programming, Elsevier. 2. Handy Taha, Operations Research – An Introduction, Prentice Hall of India, New Delhi. 3. R. Fletcher, Practical Methods of Optimization, Wiley.

CSE 616 Artificial Intelligence 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 616</td><td>Artificial Intelligence</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites
Data Structure and Algorithm, DBMS, Object Oriented Programming</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Learns Concepts of Intelligence, Artificial Intelligence, Problem Representation and Characterization.
● CO2: Conceptualizes Intelligent Search, different heuristics.
● CO3: Understands Knowledge Representation Techniques and Uncertainty Managements.
● CO4: Learns Semantic Knowledge, Semantic Net and Frame.
● CO5: Learns Game Playing Program Design.
● CO6: Learns Expert Systems and Various Machine Learning Systems.
● CO7: Learns Neural Networks.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Artificial Intelligence (AI): Features of natural intelligence, Definition of Artificial Intelligence (AI), Turing Test. (4L)

Problem Representation and Characterization: State Space Representation, Production Systems, Search, Problem Characterization. (5L)

Intelligent Search Techniques: Search Classifications, Heuristic Function, Various Types of Heuristic Search Techniques, Performance Measure of Heuristic Search with Penetrance. (5L)

Knowledge Representation Methodologies: Types of Knowledge, Propositional vs. Predicate Logic, Resolution Proof, Logic Programming, Knowledge representation using Rules, Declarative and Procedural Representation, Uncertainty Management in Knowledge Representation, Certainty Factors in facts and rules, Concept of Fuzzy Logic. (5L)

Semantic Knowledge Representation: Syntactic vs. Semantic Knowledge, examples of Semantic Knowledge, Semantic Net, Frame, OOP, Property Inheritance, Tangled Hierarchies. (4L)

Game Playing: Game Tree, Minimax Search, Search Reduction by alpha and beta cutoffs. Planning: Introduction to Planning, Goal Stack Planning, Nonlinear, Hierarchical and Reactive Planning. (4L)

Learning: Learning and Intelligence, Learning Spectrum, Various Types of Learning Techniques and Systems. (5L)</td></tr>
</table>

	Expert Systems (ES) and ES Shells: Definition of Expert Systems, Components of Expert Systems. Types of ES – Manual, Semi-automatic, and Automatic ES, Techniques of Knowledge Acquisition (KA) for ES.-- ES Shell. Advantages and disadvantages of ES Shell over ES. (5L) Neural Networks: Symbolic vs. Neural Network AI, Hofield Network, Perceptron as a model of neuron, Single and multiplayer Perceptron for classification and knowledge representation, Back propagation Network, Supervised, Reinforcement and Unsupervised Learning. (5L)
Text Books, and/or reference material	**Text Books:** 1. Artificial Intelligence -- Rich and Knight. -- Tata McGraw Hill. 2. Artificial Intelligence – A New Synthesis – Nilsson. -- Morgan Kaufmann Publishers. **Reference Books:** 1. Artificial Intelligence and Expert Systems -- Paterson. -- PHI. 2. Artificial Neural Networks – B. Yegnanarayanana. PHI.

CSE 617 Advanced Algorithms 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 617</td><td>Advanced Algorithms</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites
CSC 303, CSC 403</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1. Can have the efficiency in the complexity analysis of the algorithms.
● CO2. Detecting and applying the algorithmic structures in many different fields of engineering.
● CO3. Will have the knowledge for state of the art development in the field of algorithms.
● CO4. Can have the proficiency of coding and comparing different algorithms.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Revisit: Different Complexity analysis and Algorithm’s correctness by Loop-Invariant techniques. (2L)

Data Structures: van Emde Boas Trees, Dynamic graphs, Bloom filters, Hashing (Open addressing). (5L)

Randomized Algorithm- Las Vegas and Monte Carlo algorithms, Essential mathematical tools for Randomized algorithms: Linearity of expectation, Markov inequality, Chebyshev's inequality, Chernoff bound, and Union bound with examples to Randomized algorithm design. Examples and analysis of: Hiring Assistant Problem, Randomized selection, Skip list. (4L)

Network Flow - Flow networks, Augmenting paths, Ford- Fulkerson Algorithm, Edmonds - Karp algorithm, Max flow min-cut theorem, Push-relabel algorithm, Maximum bipartite matching, Some applications of network flow. (5L)

Linear Programming: Introduction, algorithms, and its applications, Linear programming duality. (4L)

Parallel Algorithms – Multithreaded Algorithms: Multithreaded matrix multiplication, Multithreaded merge sort. (3L)

Online Algorithms: Overview, Online scheduling and online Steiner tree, Online Bipartite matching, Online learning and multiplicative weights algorithm. (5L)

NP- Completeness - Reduction revisited; NP-Completeness proof of different problems: CLIQUE, VERTEX COVER, INDEPENDENT SET, SET COVER. (4L)

Approximation Algorithms - Constant factor approximation algorithm: VERTEX COVER and TSP; Christofides algorithm on TSP with 1.5 approximation factor; SET-COVER problem with log n factor approximation algorithm; PTAS and FPTAS, Linear programs and approximation algorithms. (7L)</td></tr>
</table>

	Semidefinite Programming: Introduction with the problem: The Maximum Cut Problem and Semidefinite Programming. (2L) **Overview of some Special Topics:** Communication complexity, Spectral graph theory, Compressive sensing . (1L)
Text Books, and/or reference material	**Text Books:** 1. Rajeev Motwani and Prabhakar Raghavan, Randomized Algorithms, 2nd Edition, Cambridge University press, Cambridge, MA, 1995. 2. Thomas H. Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein. Introduction to Algorithms. 3rd ed. MIT Press, 2009, ISBN: 9780262033848. 3. S. G. Akl, The Design and Analysis of Parallel Algorithms, Prentice-Hall, 1989. 4. M. J. Quinn, Designing Efficient Algorithms for Parallel Computers, McGraw Hill Higher Education, 1987, ISBN: 978-0070510715. 5. J. Kleinberg and E. Tardos, Algorithm Design, Pearson. 6. D. V. Williamson and D. B. Shmoys, The Design of Approximation Algorithms, Cambridge University Press. 7. S. Arora and B. barak, Computational Complexity: A Modern Approach, Cambridge University Press. **Reference Books:** 1. Dimitri P. Bertsekas and John N. Tsitsiklis, Introduction to Probability, 2nd Edition, Athena Scientific, July 2008. 2. M. Mitzenmacher and E. Upfal, Probability and Compuitng: Randomized Algorithms and Probabilistic Analysis, Cambridge University Press. 3. T. Roughgarden, CS261: A Second Course in Algorithms (Stanford University), 2016. 4. T. Roughgarden, CS168: Modern Algorithmic Toolbox (Stanford University), 2017. **Others:** *NMEICT video on:* *Design of Algorithms(http://www.nmeict.iitkgp.ac.in/Home/videoLink/10/3gp)*

CSE 618 Information Coding Theory 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 618</td><td>Information coding theory</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Probability and statistics, linear algebra, calculus.</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Determine the amount of information per symbol and information rate of a discrete memoryless source.
● CO2: Design lossless source codes for discrete memoryless source to improve the efficiency of information transmission.
● CO3: Understand various lossless data compression methods.
● CO4: Evaluate the information capacity of discrete memoryless channels and determine possible code rates to achievable on such channels.
● CO5: Apply Shannon-Hartley theorem for information transmission on Gaussian channels to determine the capacity.
● CO6: Understand some well-known error correction coding techniques.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction, Mathematical Measure of Information, Average and Mutual Information and Entropy, Properties of Entropy, Discrete memoryless sources (DMS), Extension of DMS, Markov sources, Source coding theorem, Fixed length and variable length coding, Kraft inequality, Properties of prefix codes. (8L)

Source Coding: Lossless entropy encoding, Huffman code, Huffman code applied on the symbols of extended sources, Shannon-Fano coding, efficiency calculations, Lempel-Ziv codes, arithmetic coding, Rate distortion Theory. (8L)

Lossless and lossy predictive coding and decoding, Quantization, PCM, DM, ADM, DPCM. (6L)

Channels and Channel Capacity: Discrete memoryless channel model, Binary symmetric channels and channel capacity, entropy rate and channel coding theorem, information capacity theorem. (6L)

Error correction codes: Introduction, Basic concepts of linear algebra including group, ring, field, vector space etc. (3L)

Block codes: Introduction, single parity check codes, product codes, repetition codes. (3L)

Linear Codes: Definition, encoding and decoding of linear codes, generator matrix, error detection and correction, Perfect codes, Hamming codes. (5L)

Cyclic codes: Definition, polynomials, encoding and decoding techniques, cyclic redundancy check. (3L)</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. Information Theory and Coding Hardcover by Norman Abramson, McGraw-Hill. 2. Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing) by Thomas M. Cover, Joy A. Thomas, Wiley-Blackwell. 3. Error Control Coding by Shu Lin, Daniel J. Costello, Pearson. **Reference Books:** 1. Coding and Information Theory by Steven Roman, Springer-Verlag. 2. Error Control Coding by Peter Sweeney, John Wiley & Sons.

CSE 619 Computer Graphics 3-0-0 3 Credits 3 Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE619</td><td>Computer Graphics</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Introduction to Computing</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding of Graphics Hardware, Software.
● CO2: Learning various 2D algorithms and 3D algorithms.
● CO3: Learning scan conversion - lines, circles, ellipses, filling polygons, clipping algorithms.
● CO4: Learning Solid modelling, visible surface algorithms.
● CO5: Learning Illumination and Shading Models, Plane Curves and Surfaces.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Section 1 Introduction to Computer Graphics, Graphics Application and Software, Description of some graphics devices, Active and Passive Graphics Devices, Display Technologies, LCD displays. (6L)
Section 2 Two-Dimensional Transformations and Matrices, Transformation Conventions, 2D Transformations, Rotation, Reflection, Scaling. (6L)
Section 3 Three-Dimensional Transformations Introduction, Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation. (6L)
Section 4 Filling polygons and clipping algorithms, Clipping Lines algorithms– Cyrus-Beck, Cohen-Sutherland and LiangBarsky, Clipping Polygons. (6L)
Section 5 Visible-Surface Determination Techniques, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painter’s algorithms (depth sorting), Area sub-division method, BSP trees. (6L)
Section 6 Illumination and Shading Illumination and Shading Models for Polygons, Reflectance properties of surfaces, Ambient, Specular, and Diffuse reflections, Atmospheric attenutation, Phong’s model, Gouraud shading, some examples. (6L)
Section 7 Plane Curves and Surfaces Curve Representation, Parametric Representation of a Circle, Ellipse, Parabola, Hyperbola, Space Curves, Cubic Splines, Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1) J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Computer Graphics - Principles and Practice, Second Edition in C, Pearson Education, 2003.
2) D. F. Rogers and J. A. Adams, Mathematical Elements for Computer Graphics, 2nd Edition, McGraw-Hill International Edition, 1990.
Reference Books:
1) D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education, 2nd Edition, 2004.
2) F. S. Hill Jr., Computer Graphics using OpenGL, Pearson Education, 2003.
Others:
NPTEL Course: https://nptel.ac.in/courses/106106090/</td></tr>
</table>

CSE 620 Game Theory and its Applications 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 620</td><td>Game Theory and its Applications</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT), Mid-Term (MT) end assessment (EA))</td></tr>
<tr><td colspan="3">1. MAC 01: Mathematics - I
2. MAC 02: Mathematics - II
3. MAC 331 : MAC 01: Mathematics - III
4. CSC 403: Design and Analysis of Algorithms</td><td colspan="5">CT: 15%, MT: 25%, EA: 60%</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Can have the efficiency to act in a strategic situation.
● CO2: Can analyse the strategic interactions among agents.
● CO3: Can understand modern state of the art in Game Theory.
● CO4: Will have the knowledge of related area where Game Theory can be applied.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Motivation to the course. (2L)

Non-Coperative Game Theory: Introduction to Game Theory, Extensive Form Games, Strategic Form Games, Dominat Strategy Equllibria, Pure Strategy Nash Equlibrium, Mixed Strategy Nash Equlibrium, Fixed Point Theorem and Existence of Nash Equillibrium, Computation of Nash Equillibrium, Complexity of Computing Nash Equillibrium, Matrix Games (Two Players Zero sum Games), Bayesian Games, Subgame Perfect Equlibrium. (10L)

Mechanism Design without Money: One sided and two sided matching with strict preferences, Voting theory, and Participatory democracy. (4L)

Mechanism Design with Money: Auction basics, sponsored search auctions, Revenue optimal auctions, VCG Mechanisms. (5L)

Cooperative Game Theory: Correlated Strategies and Correlated Equilibrium, Two Person Bargaining Problem, Coalitional Games, The Core, and The Shapley Value. (5L)

Repeated Games: Introduction to repeated games and its Applications. (4L)

Applications: Incentive Study in - P2P Networks, Crowdsourcing, Digital currency. (5L)</td></tr>
</table>

	Some Special Topics: Fair Division, Price of Anarchy, Scoring rules, Learning in Auction, Synergies between Machine Learning & Game Theory. (8L)
Text Books, and/or reference material	**Text Books:** 1. N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. Algorithmic Game Theory. Cambridge University Press, New York, NY, USA, 2007, ISSN: 978-0521872829. 2. M. Maschler, E. Solan, and S. Zamir. Game Theory, Cambridge University Press; 1st Edition, ISSN: 978-1107005488, 2013. 3. Y. Narahari. Game Theory and Mechanism Design. World Scientific Publishing Company Pte. Limited, 2014, ISSN: 978-9814525046. 4. T. Roughgarden, Twenty Lectures on Algorithmic Game Theory, Cambridge University Press, 2016, ISSN: 978-1316624791. **Reference Books:** 1. T. Roughgarden, CS364A: Algorithmic Game Theory Course (Stanford University), 2013. 2. T. Roughgarden, CS269I: Incentives in Computer Science Course (Stanford University), 2016. 3. S. Barman and Y. Narahari, E1:254 Game Theory Course (IISc Bangalore), 2012.

CSE 621 Digital Systems Testing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE621</td><td>Digital Systems Testing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Digital Logic Design, Computer Organisation</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To explain and exemplify basic and advanced concepts of Testing of Digital Circuits.
● CO2: To understand fault modeling and test generation.
● CO3 : To fully appreciate the need for testability measures in the design stage of circuits.
● CO4: To understand the use of built in testing measures for online testing.
● CO5: To appreciate the different testing strategies for memory based devices.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to VLSI testing and verification. Logic and Event Driven Simulation. (2L)

Fault Modeling. Single Stuck-at Fault model. Fault Collapsing. Fault Equivalence. Fault Domination. Checkpoint Theorem. (8L)
Fault Simulation. Serial, Parallel, Deductive and Concurrent. (3L)
Test Generation. Boolean Difference Method. D-Algorithm. PODEM. FAN. (8L)
Testability Analysis (3L)
Design for Testability. Adhoc approaches. Scan based Design. Random Scan. Scan FF design. LSSD. Scan-Hold FF. (8L)
Built-in Self Test. Pseudo-Random Pattern Generation. LFSR. (8L)
Memory testing. (2L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Essentials of Electronic Testing for Digital, Memory and Mixed Signal VLSI Circuits. Bushnell and Agrawal. Kluwer Academic Publishers.
2. Digital Systems Testing and Testable Design. Abramovici et.al. Jaico Publications.

Reference Books:
1. VLSI Test Principles and Architectures. LT Wang et.al. Morgan Kaufman.</td></tr>
</table>

CSE 622 Soft Computing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 622</td><td>Soft Computing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Introduction to computing, Data Structures and Analysis of Algorithms</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To familiarize with neural networks and learning methods for neural networks.
● CO2: To introduce basics of genetic algorithms and their applications in optimization and planning.
● CO3: To introduce the ideas of fuzzy sets, fuzzy logic and fuzzy inference system.
● CO4: To introduce students' tools and techniques of Soft Computing.
● CO5: To develop skills thorough understanding of the theoretical and practical
● aspects of Soft Computing.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Module I: Introduction (6L)

Introduction and different definitions of Soft Computing, Basic tools/members of Soft Computing: Fuzzy Logic, Neural Network and Evolutionary Computing.

Module II: Fuzzy Logic (10L)

Fuzzy Logic-I: Crisp Sets, Fuzzy sets, Fuzzy membership functions, Basic operations on fuzzy sets, Fuzzy relations and Composition of fuzzy relations.
Fuzzy Logic –II (Fuzzy Rules and Approximate Reasoning): Fuzzy if-then rules: M-A and TSK Rules, Fuzzification, Compositional rule of Inference/Approximate Reasoning, Defuzzification, Applications: Pattern Recognition, Fuzzy c-means Clustering and Control.

Module III: Neural Networks (10L)

Neural Networks-1 (Introduction & Architecture): Introduction to neural networks: Artificial Neuron and its model, Activation functions, Neural network architecture, learning algorithms/rules, Training and testing.
Neural Networks-II: Perceptron model: single layer and multilayer perceptron (MLP), Error back propagation, Radial basis function network (RBFN), Self-organizing map network (SOMN), Recurrent neural network, Applications of ANN.

Module IV: Evolutionary Computing (12L)

Genetic Algorithm–I: Evolutionary Computing, Basic concepts and working principle of simple GA (SGA), Genetic Operators: Selection, Crossover and Mutation, flow chart of SGA, Chromosome Encoding & Decoding, Population Initialization, Objective/fitness Function, variable length Chromosome, Applications: Travelling Salesman Problem (TSP).</td></tr>
</table>

	Genetic Algorithm–II (Multi-objective Genetic Algorithm (MOGA)): Conflicting objectives, Objective space and variable space, Domination, Pareto front, Pareto Set, NSGA-II: Non-dominated Sorting, Crowding distance operator, Applications. Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Local Search and Memetic algorithm. **Module V: Hybridization of different Soft Computing Tools** (4L)
Text Books, and/or reference material	**Text Books:** 1. S. Rajsekharanand and Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications", Prentice Hall of India. 2. N. P. Padhy, "Artificial Intelligence and Intelligent Systems", Oxford University Press. 3. G. Klir and B. Yuan, "Fuzzy sets and Fuzzy logic", Prentice Hall of India. 4. K. H. Lee., "First Course on Fuzzy Theory and Applications", Springer-Verlag. 5. G. J. Klir and T. A. Folger: Fuzzy Sets, Uncertainty, and Information, PH. 6. J. Yen and R. Langari, "Fuzzy Logic, Intelligence, Control and Information", Pearson Education. 7. D. Goldberg: Introduction to Genetic Algorithm. **Reference Books:** 1. Siman Haykin, "Neural Networks", Prentice Hall of India. 2. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", Wiley India. 3. Kumar Satish, "Neural Networks", Tata Mc. Graw Hill. 4. B. Yegnanarayana , "Artificial Neural Networks" 5. A. Konar, "Computational Intelligence", Springer. 6. Y. H. Pao: Adaptive Pattern Recognition and Neural Networks, Addison-Wesley.

CSE 623 Advanced Database Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 623</td><td>Advanced Database Systems</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Fundamentals of DBMS, Data Structures</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td colspan="3">Course Outcomes</td><td colspan="5">● CO1: Acquire knowledge about the design and application view of DBMS.
● CO2: Able to analyze query expression, specially importance of query optimization.
● CO3: To learn about design, features and operations in the field of DDBMS, OODBMS and DW.
● CO4: To learn the concept of using multimedia database as a real-life application.</td></tr>
<tr><td colspan="3">Topics Covered</td><td colspan="5">Unit-1: Comparison between different databases: Significance of Databases, Database System Applications, Advantages and Disadvantages of different Database Management systems, Comparison between DBMS, RDBMS, Distributed and Centralized DB, Introduction of various types of index structures: Primary, Secondary, Multilevel, Dynamic multilevel (B-tree and B+-tree). (5L)

Unit-2: Normalization: Functional Dependency, Anomalies in a Database, The normalization process: Conversion to first normal form, Conversion to second normal form, Conversion to third normal form, The boyce-code normal form(BCNF), Fourth Normal form and fifth normal form, normalization and database design, Denormalization, Lossless join decomposition, Dependency preservation. (6L)

Unit-3: Transaction processing: Introduction of transaction processing, advantages and disadvantages of transaction processing system, online transaction processing system, serializability and recoverability, view serializability, Transaction management in multi-database system, long duration transaction, high-performance transaction system. (5L)

Unit-4: Concurrency Control Serializability: Enforcing, Serializability by Locks, Locking Systems With Several, Lock Modes, Architecture for a Locking</td></tr>
</table>

	Scheduler Managing Hierarchies of Database Elements, Concurrency Control by Timestamps, Concurrency Control by Validation, Database recovery management. (5L) **Unit-5:** Query Optimization: Algorithm for Executing Query Operations: External sorting, Select operation, Join operation, PROJECT and set operation, Aggregate operations, Outer join, Heuristics in Query Optimization, Semantic Query Optimization, Converting Query Tree to Query Evaluation Plan, multi-query optimization and application, Efficient and extensible algorithms for multi-query optimization. (5L) **Unit-6:** Query Execution: Introduction to Physical-Query-Plan Operators, One-Pass Algorithms for Database, Operations, Nested-Loop Joins, Two-Pass Algorithms Based on Sorting, Two-Pass, Algorithms Based on Hashing, Index-Based Algorithms, Buffer Management, Parallel Algorithms for Relational Operations, Using Heuristics in Query Optimization, Basic Algorithms for Executing Query Operations. (5L) **Unit-7:** Distributed Database (DDB): Introduction of DDB, DDBMS architectures, Homogeneous and Heterogeneous databases, Distributed data storage, Advantages of Data Distribution, Disadvantages of Data Distribution Distributed transactions, Commit protocols, Availability, Concurrency control & recovery in distributed databases, Directory systems, Data Replication, Data Fragmentation. Distributed database transparency features, distribution transparency. (5L) **Unit–8:** Object Oriented DBMS(OODBMS): Overview of object: oriented paradigm, OODBMS architectural approaches, Object identity, procedures and encapsulation, Object oriented data model: relationship ,identifiers, Basic OODBMS terminology, Inheritance , Basic interface and class structure, Type hierarchies and inheritance, Type extents and persistent programming languages, OODBMS storage issues. (5L) **Unit –9:** XML Query processing: XML query languages: XML-QL, Lorel, Quilt, XQL, XQuery, and Approaches for XML query processing, Query processing on relational structure and storage schema, XML database management system. (5L) **Unit –10:** Data Warehousing: Overview of DW, Multidimensional Data Model, Dimension Modelling, OLAP Operations, Warehouse Schema (Star Schema, Snowflake Schema), Data Warehousing Architecture, Virtual Data, Metadata and Types of Metadata, OLAP Engine, Data Extraction, Data Cleaning, Loading, Refreshing. **Unit-11:** Database application: Multimedia database, Video database management: storage management for video, video preprocessing for content representation and indexing. (2L)

Text Books, and/or reference material	**Text Books:** 1. "An Introduction to Data Base Systems", C. J Date, Pearson Education. 2. "DatabaseSystem Concepts", Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw-Hill. 3. "Distributed Databases Principles & Systems", Stefano Ceri and Giuseppe Pelagatti, McGraw-Hill International Editions. **Reference Books:** 1. "Fundamentals of Database Systems", Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley.

SEVENTH SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	MSC731	Principles of Management	3-0-0	3	3
2	CSE710 --	Depth Elective – 3	3-0-0	3	3
3	CSE710 --	Depth Elective - 4	3-0-0	3	3
4	CSE710 --	Depth Elective - 5	3-0-0	3	3
5	YYO74*	Open Elective - 3	3-0-0	3	3
6	CSS751	Software Engineering Laboratory	0-0-3	1.5	3
7	CSS752	Modelling and Simulation Laboratory	0-1-3	2.5	4
8	CSS753	Vocational Training / Summer Internship and Seminar	0-0-2	1	2
9	CSS754	Project - I	0-0-3	1	3
		TOTAL	**15-1-11**	**21**	**27**

7th Semester

	Basket of Depth Elective – 3, 4, 5
CSE710	**Machine Learning**
CSE711	**Graph Theory**
CSE712	**Electronic Design Automation**
CSE713	**Natural Language Processing**
CSE714	**Data Warehousing and Data Mining**
CSE715	**Digital Image Processing**
CSE716	**Data Analytics**
CSE717	**Biometrics**
CSE718	**Cryptography and Network Security**
CSE719	**Multimedia Information Systems**
CSE720	**Cellular Automata and its Application**
CSE721	**Computational Geometry**
CSE722	**Complex Network Theory**
CSE723	**Pattern Recognition**

CSE724	Semantic Web Technology
CSE725	Human Computer Interaction

7th Semester

	Basket of Open Elective – 3
CSO741	Software Engineering
CSO742	Multimedia Technologies
CSO743	Computer Networks

CSS 751 Software Engineering Laboratory 0-0-3 1.5Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 751</td><td>Software Engineering Laboratory</td><td>PCR</td><td>0</td><td>0</td><td>3</td><td>3</td><td>1.5</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="2"></td><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Understand Control Flow Graph (CFG) and CFG based Functional Complexity of the software.
● CO2: Understand the Coverage Criteria (Statement, Branch, Decision).
● CO3: Software modelling through ERD, DFD and ERD for distinct cases.
● CO4: Unified Modelling Language based system Design and code Generation.
● CO5: Understand the basic concepts of Testing and Verification (Decision tree & graph, WBT, BBT, Unit testing).</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1) Control Flow Graph based problems (Tool: C++/Java Language Compiler).

2) ERD / DFD related problems (Tool: StarUML ER Extension or Other OpenSource Tools).

3) UML based Design problems (Tool: Rational Rose/StarUML).

4) Software Testing related Problems (Tool: Junit) - Implementation Program on Java and testing using Junit. Suggested List of Applications:1. Student Marks Analysing System, 2. online Ticket Reservation System, 3. Payroll System, 4. Course Registration System, 5. Expert Systems, 6. ATM Systems, 7. Stock Maintenance.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">References:
1. Frances E. Allen, “Control flow analysis”, Proceedings of a symposium on Compiler optimization archive, ACM SIGPlan Notices, Pages 1 – 19, 1970
2. Unified Modelling Language, Object Management Group, http://www.omg.org/spec/UML/
3. JUnit User Guide, https://junit.org/junit5/docs/current/user-guide/</td></tr>
</table>

CSS 752 Modelling and Simulation Laboratory 0-1-3 2.5Credits 4Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSS 752</td><td>Modelling and Simulation Laboratory</td><td>PCR</td><td>0</td><td>1</td><td>3</td><td>4</td><td>2.5</td></tr>
<tr><td colspan="2" rowspan="2">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous (CT) and End assessment (EA))</td></tr>
<tr><td colspan="5">CT+EA [CT: 60%, EA(Laboratory assignment + Viva Voce): 40%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Demonstrate the characteristics of mathematical modelling and Python packages.
● CO2: Understand the concepts of mathematical modelling for a problem.
● CO3: Understand the user-friendly editor of Python and various libraries for simulation of the problems.
● CO4: Developed and implement the mathematical problems using Python.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">1. Study the basic concepts of mathematical formulation for a problem.
2. Study the characteristics and packages of Python programming language.
3. Modelling and simulation of linear programming problems.
a) Graphical Method
b) Simplex Method
4. Modelling and simulation of Transportation problem.
a) Different initialization solution techniques
b) Balanced and Unbalanced
c) Degenerate problem
5. Modelling and simulation of Assignment problem.
6. Modelling and simulation of travelling salesman problem.
7. Modelling and simulation of network flow problem.
8. Modelling and simulation to find the dual of a primal problem.
9. Modelling and simulation to determine optimal strategy for a two person zero game.
a) Pure Strategy
b) Mixed strategy</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="6">Text Books:
1. Rardin, Optimization in Operation Research, Pearson Publications.
2. Handy A Taha, Operations Research – An Introduction, Prentice Hall of India, New Delhi.
3. Hillier & Lieberman, Introduction to Operations Research, TMH.</td></tr>
</table>

Depth Elective – 3, 4, 5

CSE 710 Machine Learning 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 710</td><td>Machine Learning</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Probability and Statistics, Artificial Intelligence</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding of the basic concepts, fundamental issues and challenges of machine learning.
● CO2: Comprehend the principle and techniques of supervised learning.
● CO3: Explain the basic concepts and techniques of unsupervised learning.
● CO4: Understanding of the basic concepts and challenges of reinforced learning.
● CO5: Ability to apply the concepts of machine learning in different domains.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Introduction: what is Machine Learning; Human learning and Machine learning; Well-posed learning problem; Types of Machine Learning: Supervised, Unsupervised, and Reinforcement learning; Applications, Issues, and tools of Machine Learning. (3L)
2. Concept Learning: Inductive learning hypothesis, general to specific ordering of hypothesis; FIND-S algorithm; Version space, candidate elimination algorithm; Inductive bias. (4L)
3. Bayesian Learning, Naïve Bayes Classifier, Optimal Classifier (3L)
4. Supervised learning: Classification- k-Nearest Neighbour, Decision Tree, Support vector machine. Regression- Simple and Multiple linear regression. (12L)
5. Artificial Neural Networks: Biological neuron and artificial neuron, How ANN works, Parallel distributed model of ANN; Activation functions; Perceptron, McCulloch-Pits model, ADALINE network model; Architecture of ANN- single-layer feed forward, multi-layer feed forward, competitive network, recurrent network; Backpropagation algorithm; Basic concept of deep learning. (5L)
6. Unsupervised learning: Different clustering techniques- Partitioning methods (k-means, k-medoid, etc. clustering techniques), Hierarchical methods (Agglomerative and Divisive techniques: MIN, MAX, Group average, Ward's etc. methods), and Density-based method (DBSCAN). (5L)
7. Unsupervised learning: Rule mining and Association analysis- different terminology (itemset, support count, support, association rule, confidence, etc.); Association rule mining techniques; Market-Basket analysis; Apriori principle,</td></tr>
</table>

	Apriori algorithm for frequent itemset generation, Rule generation for apriori algorithm. (5L) 8. Genetic Algorithm based Learning. (2L) 9. Reinforcement Learning: Basic concept, Model based learning, Temporal difference based learning. (3L)
Text Books, and/or reference material	**Text Books:** 1. Machine Learning by Tom Mitchell [Mc. Graw-Hill]. 2. Machine Learning by S. Dutt, S. Chandramouli, and A. K. Das [Pearson, 2019]. 3. Applied machine Learning by M. Gopal [Mc. Graw-Hill, 2018] 4. NPTEL Course materials. **Reference Books:** Introduction to Machine Learning by Ethem Alpaydin [MIT Press].

CSE 711 Graph Theory 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 711</td><td>Graph Theory</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Discrete Mathematics and Data Structures</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the basic concept of graph and its properties.
● CO2: Apply the basic properties of graph theory to prove different.
● CO3: Discuss about chromatic characteristics and planar graph.
● CO4: Students can explore knowledge of graph theory to solve the technology driven and research oriented problems.
● CO5: Solve various graph theory problems.
● CO6: Use a combination of theoretical knowledge and mathematical thinking to solve various computer science applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Preliminaries: Graphs, isomorphism, automorphism, components, sub-graphs, degree, operations on graphs, radius, diameter, bipartite graph, Operations on graph: deletion of vertex/edge, fusion, union, intersection, ring sum, decomposition, join, Cartesian product, complement. Self-complementary graphs, circuits. (8L)
Connected graphs and shortest paths: Walks, trails, paths, connected graphs, distance, cut-vertices, cut-edges, connectivity: edge and vertex connectivity, relationship between edge and vertex connectivity, k-connected graph, Menger’s theorem, separable graph, blocks, block-cut vertex tree, block tree, cut vertex tree,1-isomorphism, 2-isomorphism, topological ordering. (8L)
Trees: Characterizations, number of trees, minimum spanning trees, Distance between spanning tree of a connected graph, eccentricity, Centre(s) of trees and connected graph, diameter of tree and connected graph , nullity of tree, labelled graph. (3L)
Planarity: Planar graph, Kuratowski’s theorem , Euler’s formula, Detection of planarity , duality, uniqueness of duality, Homomorphism: subdivision, merging, planarity detection using homeomorphism graphs, five color and four color problem. (5L)
Covering, Independent sets, Dominating Set, Matching: Basic concepts, vertex and edge covering, minimal covering, independent set, maximal independent set, relationship between covering and independent set, theorems, dominating set, MDS, CDS, matching in bipartite graphs, perfect matching, maximal matching, minimum matching, Hall’s theorem. (6L)
Factorization : Factor, 1-factor, 2-factor Tutte’s theorem. (3L)</td></tr>
</table>

	Vertex coloring: Chromatic number and cliques, greedy coloring algorithm, Brook's theorem, chromatic partition, Uniquely colourable graph. (3L) **Edge coloring:** Gupta-Vizing theorem, color edge, equitable edge-coloring. (2L) **Line Graph:** Properties and proof. (2L) **Eulerian graphs:** Characterization, Arbitrarily traceable graph, Fleury's algorithm. (2L)
Text Books, and/or reference material	**Text Books:** 1. Douglas B. West. Introduction to Graph Theory. Pearson Education, Second Edition. 2. R. Deistel. Graph Theory. Springer- Verlag NewYork 1997. 3. R.J. Wilson and J.J. Watkins. Graphs : An Introductory Approach. John Wiley and Sons Inc. **Reference Books:** 1. N. Deo. Graph Theory; With Applications to Engineering and Computer Science. PHI. 2. S. Pirzada. An Introduction to Graph Theory. Orient Blackswan.

CSE 712 Electronic Design Automation 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE712</td><td>Electronic Design Automation</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Digital Electronics, Computer Organisation, Algorithm Analysis and Design.</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To visit the various stages of the VLSI design cycle and appreciate the role of automation therein.
● CO2: To appreciate how High Level Synthesis converts an HDL code into an architecture level design.
● CO3: To discuss the algorithmic approach to physical design.
● CO4: To emphasize the importance to testability measures in the design.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">VLSI Design cycle. Design styles. System packaging styles. Fabrication of VLSI devices. Design rules-overview. (3L)
HLS: Scheduling in High Level Synthesis. ASAP and ALAP schedules. Time constrained and Resource constrained scheduling. (4L)
HLS: Allocation and Binding. Datapath Architectures and Allocation tasks. (4L)
Partitioning. Clustering techniques. Group Migration algorithms. (4L)
Floorplanning. Constraint based Floorplanning. Rectangular Dualization. Hierarchical Tree based methods. Simulated Evolution approaches. Timing Driven floorplanning. (5L)
Placement. Simulation based placement algorithms. Partitioning based placement algorithms. Cluster Growth. (5L)
Global Routing. Maze Routing algorithms. Line probe algorithms. Shortest Path based algorithms. Steiner’s Tree based algorithms. (5L)
Detailed Routing. Channel Routing Algorithms. Switchbox Routing. Over-the-cell routing. Clock and Power Routing. (4L)
Design for testability. Fault testing. Ad-hoc and structured DFT techniques. (8L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Algorithms for VLSI Physical Design Automation. N.A.Sherwani. Kluwer Academic Publishers.
2. High-Level Synthesis: Introduction to Chip and System Design. Gajski et. al. . Kluwer Academic Publishers.
3. Digital Systems Testing and Testable Design. Abramovici et.al. Jaico Publications.</td></tr>
</table>

	Reference Books 1. VLSI Physical Design Automation. Sadiq M. Sait and Habib Youssef. Kluwer Academic Publishers. 2. Algorithms for VLSI Design Automation. Sabih H. Gerez. Wiley India. 3. Essentials of Electronic Testing for Digital, Memory and Mixed Signal VLSI Circuits. Bushnell and Agrawal. Kluwer Academic Publishers.

CSE 713 Natural Language Processing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 713</td><td>Natural Language Processing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">1. Basics of probability and statistics
2. CSC303: Data Structures and Algorithms
3. CSC 01: Introduction to Computing</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Knowing the fundamental concepts underlying natural language processing (NLP).
● CO2: Understanding morphology, tokenization and stemming.
● CO3: Understanding the approaches to syntax and semantics in NLP.
● CO4: Understanding some NLP applications.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to natural language processing. (1L)
Basic Text Processing: Tokenization, Stemming. (2L)
Minimum Edit Distance. (2L)
Language Modeling: Introduction to N-grams, Estimating N-grams probabilities. Application of language modeling to real-life examples (such as text -classification). (5L)
Generative Vs. Discriminative Models. (4L)
POS Tagging. (4L)
Parsing: Introduction of Probabilistic Parsing, Lexicalized Parsing, Dependency Parsing. (6L)
Information Retrieval. (3L)
Semantics: Word meaning and Senses. (3L)
Machine Translation (rule based techniques, Statistical Machine Translation (SMT), parameter learning in SMT (IBM models)). (5L)
Two applications: Question Answering and Text Summarization. (4L)
Recent trends. (3L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

Jurafsky, David, and James H. Martin. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. Prentice-Hall, 2000. ISBN: 0130950696.

Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.</td></tr>
</table>

	Reference Books: Manning, Christopher D., and Hinrich Schütze. Foundations of Statistical Natural Language Processing. Cambridge, MA: MIT Press, 1999. ISBN: 0262133601. **Others:** 1. CS124: YouTube lecture videos by Dan Jurafsky. 2. 2012 NLP MOOC by Dan Jurafsky with Chris Manning:Youtube channel lecture videos

CSE 714 Data Warehousing and Data Mining 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="7">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 714</td><td>Data Warehousing and Data Mining</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites
Artificial Intelligence, DBMS, Object Oriented Programming</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">● CO1: Understanding the Concept of Data Warehousing and Data Mining.
● CO2: Association Rules: Item set, Support, Confidence.
● CO3: Classification – Pattern: Labelled Pattern, Decision Trees.
● CO4: To understand the SVM, Generalization Error.
● CO5: To understand the different types of Clustering Methods.
● CO6: To understand the detection of different types of outliers and outlier detection.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">Data Warehousing: Multidimensional Data Model, Dimension Modelling, OLAP Operations, Slicing and Dicing, Warehouse Schema, Star Schema, Snowflake Schema, Advantages and Disadvantages of Snowflake Schema, Data Warehousing Architecture, Virtual Data Warehouse, Advantages and Disadvantages of Virtual Data Warehouse, Metadata, Types of Metadata, OLAP Engine, Different Options for OLAP Engine, Data Extraction, Data Cleaning, Loading, Refreshing. [4L]

Data Mining: Different Definitions of Data Mining, KDD vs. Data Mining, Stages of KDD , DBMS vs. DM, AI vs. DM, Classifications of Data Mining, Stages of KDD, DM Techniques , Discovery Driven Tasks, Classification, Frequent Episodes, Discovery of Association Rules , Clustering, Deviation Detection, Mining Problems, Applications of DM, Other Mining Problems. [4L]

Association Rules: Item set, Support, Confidence, Problem Decomposition, Frequent Item Set, Maximal Frequent Set, Border Set, Applications of Data Mining, Spotting Fraudulent Behaviour, Astronomy etc., Association Rules, Informal a priori Algorithm for Learning Association Rules, Finding Frequent Sets and Association Rules, Formal a priori Algorithm for Association Rule. [5L]

Classification – Pattern: Labelled Pattern, Approaches of Classification, Evaluation of Classifiers, Normalized Confusion Matrix, Accuracy, Precision, Recall and F – score, Cross Validation Technique, Classification Techniques. [4L]

Decision Trees: Inductive Learning, ID3 Program, Algorithm for Building Decision Trees , Advantages of Decision Trees for Classification Purpose, Development of Decision Trees for Different Training Data Sets, Rule Extraction from Pattern Set, Covering the instances, Extraction of rules, Instance Space, Covering Algorithm. [4L]</td></tr>
</table>

	Bayesian Belief Nets (DAG): K nearest Neighbour, ANN, Learning in ANN, Perceptron as a model of neuron, Single and multiplayer Perceptron for classification and knowledge representation, Back propagation Network, Supervised, Reinforcement and Unsupervised Learning. [4L] Classification (Complex): Support Vector Machine (SVM), Generalization Error, SVM to find out the best classification, Margin. [3L] Clustering: Partitioned and Hierarchical Clustering, k means Clustering, Fast k Means Clustering, Fuzzy K means Clustering, Hierarchical Clustering, Agglomerative and Divisive Hierarchical Clustering, Single Linkage, Complete Linkage and Average Linkage Clustering. [4L] Clustering (Complex): Outlier Detection, Outlier vs. Cluster, Types of Outliers, Outlier Detection Methodologies, Supervised, Unsupervised and Semi supervised detection , Statistical Approaches, Parametric and Non Parametric Methods, Proximity Based Methods, Clustering Based Methods. [4L] Temporal and Spatial Data Mining: Temporal Data Mining, Tasks involved, Temporal Association Rules, Sequence Mining, Episode Discovery, Spatial Mining, Tasks involved , Spatial Clustering. [3] Web Mining: Web Mining Techniques, Web Content Mining, Web Structure Mining, Web Usage Mining, Text Mining. [3L]
Text Books, and/or reference material	**Text Books:** 1. Data Mining Techniques – Arun K Pujari – Universities Press. 2. Data Mining – Vikram Pudi, P. Radha Krishna – Oxford University Press. **Reference Books:** 1. Data Mining – J. Han, M. Kamber, J. Pei -- Elesvier. 2. Data Mining – Hand, Mannila and Smith – PHI.

CSE 715 Digital Image Processing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 715</td><td>Digital Image Processing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">NIL</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Acquire knowledge about image acquisition and camera basics.
● CO2: To learn the basic algorithms on filtering, quality metrics, segmentation.
● CO3: To learn about compression and color image processing.
● CO4: Development of image processing programs using ImageJ and Python.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction, Image acquisition process, image sensors, camera basics. (4L)

Transform functions, Histogram, spatial and frequency filtering. (8L)

Redundancy, compression models, coding methods. (8L)

Point, Line, edge detection, thresholding, region based segmentation. (6L)

Color models, color image processing, segmentation and compression using colors. (8L)

Introduction to Image Processing using ImageJ and Python, Image databases. (8L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Digital Image Processing by Rafael C Gonzalez & Richard E Woods.
2. Fundamentals of Digital Image Processing by Anil K Jain.

Reference Books:
Digital Image Processing by William K Pratt.
Others:
NPTEL online course.</td></tr>
</table>

CSE 716 Data Analytics 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 716</td><td>Data Analytics</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Knowledge in handling and analyzing extremely large datasets.
● CO2: Learns the techniques of uncovering hidden patterns, correlations and other insights out of these datasets.
● CO3: Ability to apply the concepts of data analytics in different domains.
● CO4: Ability to contextually integrate and correlate large amounts of information.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Data Analytics, Types of Data Analytics: Descriptive Analytics, Diagnostic Analytics, Predictive Analytics, and Prescriptive Analytics. Use Cases, Issues and Challenges in Big Data Analytics. (4L)

Fundamentals of Statistics: Population, Sample, Parameter, Statistic, Variable. Descriptive Statistics, Inferential Statistics. Basic Probability Theory: Random Experiment, Sample Space, Random Variables, Probability, Conditional Probability, Independence, Conditional
Independence, Expectation, Variance, Probability Distribution, Joint Probability Distribution, Conditional Probability Distribution. (8L)

Similarity Measures: Jaccard Similarity, Cosine Similarity, Adjusted Cosine Similarity. Missing Value Prediction Techniques: Mean Centering, Weighted Average, Z-Score. (6L)

Basics of Complex Network: Scale-Free Networks, Small-World Phenomenon, Degree Distributions, Transitivity or Clustering. Centrality Measures: Degree Centrality, Betweenness Centrality, Closeness Centrality, Eigenvector Centrality, PageRank Centrality. Community Structure, Community Detection Algorithms: Girvan-Newman, Fast Greedy, Label Propagation, Clique Percolation Method. Community Quality Metrics: Modularity, NMI, Conductance. (10L)

Introduction to Data Mining, Machine Learning Techniques: Least Square Regression, Decision-trees, SVM. Clustering Techniques: K-Means. (8L)

Introduction to Hadoop Ecosystem – HDFS, Map-Reduce, PIG, HIVE, HBase, Mahout, Zookeeper, Flume, Sqoop, etc. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

1. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data – EMC Education Services – Wiley.
2. Machine Learning: Hands-On for Developers and Technical Professionals – Jason Bell – Wiley.</td></tr>
</table>

	Reference Books: 1. Networks: An Introduction – M. E. J. Newman – Oxford University Press. 2. Hadoop: The Definitive Guide – Tom White – O'Reilly.

CSE 717 Biometrics 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 717</td><td>Biometrics</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Basic Mathematics – Knowledge and ability to use calculus, probability, and statistics are essential.</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding biometrics systems and its different characteristics.
● CO2: Implementation of different biometrics systems including face, fingerprint, iris, palm, signature, EEG, etc.
● CO3: Apply the concept of unimodal and multimodal paradigms in biometrics systems.
● CO4: Analyze different feature extraction and learning techniques for biometrics systems.
● CO5: Design and develop real life biometrics systems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Biometrics Overview: Introduction, characteristics of biometric systems, biometric systems, biometric functionalities, biometrics system errors, design cycles of biometric systems, applications of biometric systems, security and privacy issues. [4L]

Image Processing Techniques: What is image processing?, origin of image processing, fundamental steps in digital image processing, components of image processing system, image sensing and acquisition, image sampling and quantization, basic relationships between pixels. [6L]

Filtering: Background, basic intensity transformation functions, histogram processing, fundamentals of spatial and frequency domain filtering, smoothing filters, sharpening filters, Discrete Fourier Transform, Fast Fourier Transform. [4L]

Pattern Classification Techniques: Introduction, Bayesian decision theory, maximum likelihood and Bayesian parameter estimation, non-parametric techniques, linear discriminant functions, multilayer neural networks, non-metric methods. [6L]

Fingerprint Recognition: Introduction, ridge pattern, fingerprint acquisition, feature extraction, matching, and fingerprint synthesis. [6L]
Face Recognition: Introduction, image acquisition, face detection, feature extraction, matching and advanced topics. [6L]

Iris Recognition: Introduction, iris recognition systems, image acquisition, iris segmentation, iris normalization, iris encoding and matching, iris quality and performance evaluation. [4L]

Multi-modal Biometric Systems: Introduction, sources of multiple evidence, acquisition and processing architecture, fusion levels. [2L]</td></tr>
</table>

	Other Biometrics: Signature, hand shape, ear, palmprint, etc. [4L]
Text Books, and/or reference material	**Text Books:** ● Anil K. Jain, Arun Ross, and Karthik Nandakumar, Introduction to Biometrics, Springer, 2011. ● J. L. Wayman, Anil K. Jain, D. Maltoni, D. Maio, Biometric Systems: Technology, Design and Performance Evaluation, Springer, 2005. ● R. M. Bolle, J. Connell, S. Pankanti, N. K. Ratha, A. W. Senior, Guide to Biometrics, Springer, 2004. ● Richard O. Duda, Peter E. Hart, David G. Stork, Pattern Classification, 2nd Edition, Wiley, 2000. ● R.C. Gonzalez and R. E. Woods, Digital Image Processing, Pearson, 2009. **Reference Books:** ● D. R. Kisku, P. Gupta and M. Tistarelli, Multibiometrics Systems: Modern Perspectives to Identity Verification, LAMBERT Publishing, 2012. ● D. R. Kisku, P. Gupta and J. K. Sing, Advances in Biometrics for Secure Human Authentication and Recognition, CRC Press, Taylor & Francis, 2013. ● D. R. Kisku, P. Gupta and J. K. Sing, Design and Implementation of Healthcare Biometric Systems, IGI Global, 2019. ● M. Dawson, D. R. Kisku, P. Gupta, J. K. Sing and W. Li, Developing Next-Generation Countermeasures for Homeland Security Threat Prevention, IGI Global, 2016. **Others:** Online Biometrics Courses 1. https://nptel.ac.in/courses/106104119/ 2. https://www.mooc-list.com/tags/biometric

CSE 718 Cryptography and Network Security 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 718</td><td>Cryptography and Network Security</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">CSE 602
Basic knowledge of linear algebra, probability theory. Programming skills are desirable.</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Introduce to the basic mechanisms of Cryptography.
● CO2: Notion of computationally hard problems and their applications.
● CO3: Notion of trap-door and one-way functions and their applications.
● CO4: The attack and crypto-analysis.
● CO5: Ability to design secure protocols and their vulnerability analysis.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Introduction, X.800 : Security architecture for Open Systems Interconnection, Attack, Adversarial Behavior. (2L)
2. Basic Number Theory, Field, Extension Field and applications. (4L)
3. Confidentiality, Symmetric and Asymmetric Encryption, Public key encryption mechanisms - RSA, ElGamal, Rabin's, Asymmetric Key Encryption - DES, AES. (8L)
4. Attacks- Passive attacks, Side channel Attacks, Factorizations and Index calculation methods, Countermeasures. (6L)
5. Implementational Issues - Fast Hardware for symmetric and Asymmetric key. (4L)
6. Pseudo-random number generation, Stream ciphers. (2L)
7. Message Integrity, Cryptographic hashing , Message Authenticity, Message Authentication code. (2L)
8. Entity Authentication, Digital signature, Nonrepudiation. (4L)
9. Secure protocol designing - SSL, PGP and TLS. (2L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Handbook of Applied Cryptography, CRC Press (free ebook).
2. Douglas Robert Stinson, Maura Paterson, Cryptography: Theory and Practice.
3. O. Goldrich, Fundamentals of Cryptography: Basic Tools, Cambridge University Press.
4. N. Koblitz, A Course in Number Theory and Cryptography.
5. Abhijit Das, C. E. Veni Madhavan, Public-Key Cryptography: Theory and Practice.
Reference Books:
1. M. Bellare and S. Goldwasser, Lecture Notes on Cryptography, 2001.
2. Abhijit Das, Computational Number Theory, CRC Press.
Others:
1. Janathan Knudsen, Java Cryptography, O'Relly Press.</td></tr>
</table>

CSE 719 Multimedia Information Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 719</td><td>Multimedia Information Systems</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Knowledge of data structures databases and compression techniques</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">•CO1: In depth understanding of media characteristics and resource requirement.
•CO2: Organizing multimedia content, physical storage and retrieval of multimedia data, Content-based Search and retrieval, creating and delivering networked and multimedia presentations, securing multimedia content and current research directions in this area.
•CO3: Understanding networking of multimedia data and how technology can help us access, deliver, browse, search, enrich and share multimedia content.
•CO4: Understanding of multimedia database storage and retrieval.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Overview of multimedia system: Textual information codes (Morse, ASCII, EBCDIC), audio, video and graphics, RTF, TIFF, RIFF. (3L)
Video and Animation: Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options. Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Virtual Reality, Video signal representation, Computer Video Format, Computer- Based animation, Animation Languages, Methods of controlling Animation, Display of Animation, Transmission of Animation. (10L)
Information representation, media synchronisation, SAS factors, relative and absolute temporal specifications, networking delays, Skew, Jitter, end to end delay factors, latency time for stored and captured objects. (6L)
Data Compression: Storage Space requirement, Coding Requirements Source, Entropy Coding Lossy Sequential DCT- based Mode, Expanded Lossy DCT-based Mode, JPEG and MPEG. (8L)
Data transmission techniques like simplex, duplex, baseband vs. broadband, synchronous transmission vs. asynchronous transmission, synchronization parameters. (5L)
Content-based Search and retrieval, creating and delivering networked and multimedia presentations, storage, manipulation, and retrieval of multimedia data residing across global computer networks, multimedia databases, indexing, retrieval by similarity. (10L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

Multimedia Information Networking, Nalin K.Sharda, Prentice Hall India.
Multimedia: Computing, Communications and Applications, Ralf Steinmetz and Klara Nahrstedt, Pearson Education Asia.
Multimedia Communications, Applications, Networks, Protocols and Standards, Fred Halsall, Pearson Education Asia.</td></tr>
</table>

	Multimedia Systems, John F. Koegel Buford, Pearson Education Asia. **Reference Books:** Subrahmanian and Jajodia, Multimedia Database Systems, Springer. V.S. Subrahmanian, Principles of Multimedia Database Systems, Morgan Kaufmann Publishers, 1998.

CSE 720 Cellular Automata and its Application 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE720</td><td>Cellular Automata and its Application</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Digital Electronics</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding the basic and advanced concepts of Cellular Automata (CA).
● CO2: Understanding the different phases of evolution of CA machine.
● CO3: Understanding the method of characterization of CA machine/tool.
● CO4: Modeling of physical/real-time systems with a mathematical tool such as CA. Applying suitable class of CA for building CA based model to study.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Basic definitions of cellular automata and symbolic dynamics, Injectivity, surjectivity, reversibility, Garden-of-Eden theorem, Hedlund’s theorem, Conservation laws, universal computing reversible/irreversible CA, neighbourhood, dimensions, states, follow-up and review. [6L]

Characterization of CA behaviour and its applications: Initial Phase of Development, CA-Based Models - Language Recognizer, Biological Applications, CA as Parallel and Image Processing Systems, CA based model of physical systems. [6L]

New Phase of Development–Wolfram’s model of CA, 3-neighborhood 2-state CA, CA rules, Classification of rules, CA technology, CA as an FSM, Linear/non-linear/additive CA, Polynomial Algebraic Characterization of CA Behavior, Matrix Algebraic Characterization. [6L]

Irreversible/Group CA characterization in linear domain: Null/Periodic boundary Characterization of the State-Transition Behavior, Cycle Set Characterization, Isomorphism between a CA and an LFSR. CA based Pseudorandom Pattern Generation, Pseudo noise sequence, CABIST, Pattern Classification. [6L]

Characterization of nongroup CA/non-invertible CA in linear domain: General Characterization of Cyclic States (attractors), Characterization of Single Length Cycle Single Attractor CA (SACA), D1*CA, Multiple-Attractor Cellular Automata (MACA)[6L]

Non-linear CA: Characterization of non-linear rules, invertible and non-invertible CA, CA with point states; applications in VLSI domain. [6L]

Advanced Concepts: Extension of dimension, d-state CA, introduction to Asynchronous CA, follow-up and review. [6L]</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. Additive Cellular Automata: Theory and Applications, by Parimal Pal Chaudhuri, Dipanwita Roy Chowdhury, Sukumar Nandi, Santanu Chattopadhyay, Wiley. 2. Tommaso Toffoli, Norman Margolus. Cellular Automata Machines: A New Environment for Modelling. MIT Press. 3. *Cellular Automata and Complexity: Collected Papers* by Stephen Wolfram; Westview Press. **Reference Books:** 1. Game of Life Cellular Automata, by Andrew Adamatzky, Springer; 2010 edition. 2. A New Kind of Science, by Stephen Wolfram, Wolfram Media. 3. A New Kind of Computational Biology, by Chaudhuri, P.P., Ghosh, S., Dutta, A., Choudhury, S.P; Springer. 4. Joel L. Schife. Cellular Automata: A Discrete View of the World. Wiley - Interscience.

CSE 721 Computational Geometry 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 721</td><td>Computational Geometry</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">A course on Design and analysis of algorithm</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To demonstrate familiarity with some of the basic algorithmic techniques of the area.
● CO2: To design and analyze “new” geometric algorithms.
● CO3: To map practical problems to computational geometric problems and finding a solution to these geometric problems help to solve a wide range of practical problems in a variety of fields such as graphics, robotics, databases, sensor network
● CO4: To derive the lower bound for some geometric problems.
● CO5: To develop skills to work on geometrical manipulating software.
● CO6: To demonstrate acquaintance with modern research in the field.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Computational Geometry Introduction: Historical perspectives, Geometric preliminaries, Convex Hull, Algorithms to find the Convex Hull of a point set in 2D plane: Graham’s Scan Algorithm, Divide and Conquer algorithm, Output sensitive algorithm: Jarvis’s March Algorithm, Timothy Chan’s Algorithm; Lower bound analysis for Convex Hull Algorithm. [6L]

Line Segment Intersection: Line Segment Intersection, The Doubly-Connected Edge List, Computing the Overlay of Two Subdivisions, Boolean Operations. [4L]

Polygon Triangulation: Guarding and Triangulations, Area of a simple polygon, Counting the number of triangulations in a convex polygon, Art Gallery Theorem, Monotone Polygon, Partitioning a Polygon into Monotone Pieces, Triangulating a Monotone Polygon. [6L]

Orthogonal Range Searching: 1-Dimensional Range Searching, Kd Trees, Range Trees, Higher-Dimensional Range Trees, Fractional Cascading. [6L]

Point Location: Point Location and Trapezoidal Maps, A Randomized Incremental Algorithm to compute a Trapezoidal Map and a Search structure, Kirkpatrick’s planar point location problem. [7L]

Voronoi Diagram and Delaunay Triangulation: Definition and Basic Properties of Voronoi Diagram, Computing the Voronoi Diagram: Fortune Sweep Algorithm, Divide and Conquer Algorithm. Closest pair Problems. Application of voronoi diagrams, Triangulations of Planar Point Sets, The Delaunay Triangulation, Computing the Delaunay Triangulation. [8L]</td></tr>
</table>

	Arrangements and Duality: Arrangement of lines, Zone theorem, Duality, Application of arrangements and duality, Ham Sandwich Cut. [4L] **Geometric Data Structure:** Interval Trees, Priority Search Trees. [3L]
Text Books, and/or reference material	**Text Books:** 1. Franco P. Preparata and Michael Ian Shamos, Computational Geometry- An Introduction, Springer Verlag. 2. Mark de Berg, Marc van Kreveld, Mark Overmars, Otfried Cheong, Computational Geometry: Algorithms and Applications, Third Edition, Springer Verlag. 3. Joseph O' Rourke, Computational Geometry in C, Cambridge University Press. **Reference Books:** **Others:** Lecture notes on Computational geometry by David Mount.

CSE 722 Complex Network Theory 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 722</td><td>Complex Network Theory</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Probability, Calculus, Linear Algebra, Graph Theory</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: A broad conceptual introduction to the modern theory and applications of network science.
● CO2: Understand structure of communities in different networks like social networks.
● CO3: Understand random walk and its real-world applications like page ranking algorithm.
● CO4: Application of linear algebra and probability to real-world complex network problems.
● CO5: Cultivate reading of research papers and articles.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">· Introduction to Network Science (2L)
· Graph Theory: revision of basic concepts. (2L)
· Properties of Complex networks: Degree distribution, associativity, clustering coefficient.(5L)
· Random Networks: Poisson’s distribution, giant component and its emergence, generating function, component size distribution. (8L)
· Bipartite networks: unipartite projection, giant component condition. (5L)
· Centrality measures: degree centrality, closeness centrality, betweenness centrality, eigen vector centrality, Peron Frobenius theorem.(5L)
· Spectral Graph Theory: eigen values and eigen vectors, spectrum of a graph, spectrum of a clique, eigen values and eigen vectors of special matrices like triangular and diagonal matrices, Markov matrix, trace of a matrix, physical interpretation of principal eigen vector, spectral coverage, significance of 2nd eigen vector, Motifs, Frobenius norms, dimension reduction. (5L)
· Network Models: Erdos Renii graph, power law distribution in small world network, scale free networks. (4L)
· Random walks on graphs and its applications: random walks and Markov chain, transitional probability, stationery state, hitting time, commute time, cover time, mixing rate, stochastic matrix, page rank algorithm, page rank ++, HITS (Hypertext induced topic selection) algorithm by Klienberg, HITS on citation networks, bibliographic coupling, SALSA (The stochastic approach to Link Structure analysis and TKC effects). (10L)
· Community detection algorithms: what is a community, core community, Wu-Huberman Algorithm, Radicchi’s Algorithm, community detection algorithms based on shortest path betweenness and random walk betweenness. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
● “ The structure and dynamics of networks” by Newman, Barabasi, Watts, Princeton University Press.
● “Networks: An Introduction” by Mark Newmann, Oxford University Press</td></tr>
</table>

	● "Network Science" by Barabasi, Cambridge University Press. **Reference Books:** ● "Network Science" Theory and Applications by Ted G Lewis, Wiley. **Others:** ● http://www.infocobuild.com/education/audio-video-courses/computer-science/complex-network-theory-iit-kharagpur.html (Video Lecture) by Dr. Animesh Mukherjee

CSE 723 Pattern Recognition 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 723</td><td>Pattern Recognition</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites
Artificial Intelligence, Data Mining, DBMS, Object Oriented Programming</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Idea about Pattern and Pattern Class, Design of a Pattern Recognition System.
● CO2: Idea of Instar, Outstar, Groups of Instar and Outstar, Different types of Memories.
● CO3: Concept of Feedforward, Feedback and Competitive Learning Network.
● CO4: Concept of Complex PR Tasks: RBF, RBF Network for Pattern Classification.
● CO5 : Idea of Temporal Pattern Recognition: Concepts.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Pattern and Pattern Class: Design of a Pattern Recognition System, Syntactic and Decision Theoretic Approach, Bayesian Decision Theory, Continuous Features, Error, Risk and Loss. [4L]

Parametric and Non Parametric Methods: Histogram Method – Kernel Based Methods – K - Nearest Neighbor Method -- Probabilistic Neural Network base on Parzon Window – PNN Learning. [3L]

Basics of ANN: Instar , Outstar, Groups of Instar and Outstar, Different types of Memories. [3L]

PR Tasks: PR Problems, Different PR Tasks by FF, FB and Competitive Learning Network, Pattern Clustering, Feature Mapping Problem, Different Feature Mapping Network, Self Organizing Network. [4L]

FF ANN: Pattern Association Network, Hebb's Law, Pattern Classification Network. [3L]

FB ANN: Pattern Association, Pattern Storage, Pattern Environment Storage, Auto association , Hopfield Network, Capacity and Energy of a Hopfield Network, State Transition Diagram, Stochastic Network and Boltzmann Machine. [5L]

Competitive Learning Network: Pattern Storage, Pattern Clustering Network, Minimal Learning, Malsburg Learning and Leaky Learning. [4L]

Complex PR Tasks: RBF, RBF Network for Pattern Classification, Advantages of RBF over MLFF ANN, CPN Network. [3L]</td></tr>
</table>

	Single and Multilayer Network: Gradient Descent Procedure, Newton's Algorithm, Fixed Increment Learning, Variable Increment Learning, Support Vector Machine(SVM), Multilayer Neural Networks, Unsupervised Learning. [5L] Temporal Pattern Recognition: Concepts, Problems in temporal sequence, Architecture for temporal PR Tasks, Avalanche Structure, Jordon Network, Fully Connected Recurrent Network, Difference between Avalanche Network and Jordon Network. [4L] Similarity Measures: Mahalanabis Distance, Properties of Metrics, Minkowski Metric, Manhattan / City Block / L1 norm, Euclidean Distance L2 Norm, Maximum Value Distance L∞ Norm, Hamming Distance L1 norm. [4L]
Text Books, and/or reference material	**Text Books:** 1. Pattern Classification – Duda, Hart & Stork – J. Wiley & Sons. 2. Artificial Neural Networks – B. Yegnanarayana – PHI. **Reference Books:** 1. Neural Networks for Pattern Recognition – C.M. Bishop – Oxford.

CSE 724 Semantic Web Technology 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE-724</td><td>Semantic Web</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Data structure, DBMS, Web Technology, Basic Computer Logic</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">CO1: Students can write their own semantic web page by using publicly available vocabulary.
CO2: Students can publish their data in Open Data format, such that the other people can discover it easily.
CO3: Students can able to develop semantic web application.
CO4: Students will get exposure in this topic for further higher studies and research.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Principles of Linked Data, Introduction, A Layered Approach. (3L)
Naming Things with URIs, Making URIs Dereferenceable. (3L)
The Semantic Web (SW) vision: What is SW? The difference between Current web and SW, SW technologies, the Layered approach. (5L)
The XML Language, Structuring, Namespaces, Addressing and Querying XML Documents. (5L)
Resource Description Framework, RDF syntax, RDF Schema (RDFS). (7L)
Construction RDF and RDFS: Different syntax implementation, How to Store into server, Construction of RDFS. (5L)
SPARQL: Query Language: Syntax and Query processing. (2L)
Web Ontology Language OWL: OWL Syntax and Intuitive Semantics, OWL Species. (4L)
Description Logics, Model-Theoretic Semantics of OWL. (4L)
Ontology Engineering: Introduction, Constructing Ontologies, Reusing existing Ontologies. (2L)
Protégé tools. (2L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Semantic Web Primer: second edition by Grigoris Antoniou and Frank van Harmelen.
2. Foundations of Semantic Web Technologies by Hitzler Pascal.
Reference Books:
1. Ontological Engineering by Asunción Gómez-Pérez, Mariano Fernández-López, and Oscar Corcho.
2. Linked Data: Evolving the Web into a Global Data Space by Tom Heath and Christian Bizer.
Others:
Harald Sack semantic web videos.</td></tr>
</table>

CSE 725 Human Computer Interaction 3-0-0 3 Credits 3 Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 725</td><td>Human Computer Interaction</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">NIL</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Acquire knowledge about Components of HCI.
● CO2: To learn the basic Psychology of Usable Things.
● CO3: To learn about Usability Engineering, Usability Benchmarking.
● CO4: To learn Inspection methods, testing methods, design.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction, Psychology of Usable Things. (7L)
Usability Engineering, Know the User, Usability Benchmarking. (7L)
Goal-Oriented Interaction Design, Prototyping. (7L)
Usability Inspection Methods, Usability Testing Methods. (7L)
Usability in Practice, Visual Design and Typography. (7L)
Icon Design, Case Studies. (7L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Dix A., Finlay J., Abowd G. D. and Beale R. Human Computer Interaction, Pearson Education, 2005.
2. Preece J., Rogers Y., Sharp H., Baniyon D., Holland S. and Carey T. Human. ComputerInteraction, Addison-Wesley, 1994.
Reference Books:
B. Shneiderman, Designing the User Interface, Addison Wesley 2000.
Others:
NPTEL online course.</td></tr>
</table>

Open Elective – 3

CSO741 Software Engineering 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO741</td><td>Software Engineering</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Identify and describe software life cycle model and their roles in building software project.
● CO2: Recognize the feasibility of functional and non-functional requirements applying decision tree/table minimization techniques/methodologies for a particular problem.
● CO3: Apply modularity in project resulting design of flexible software code with reusability.
● CO4: Effectively use existing testing strategy to test the software and make sure the reliability of the software and analysis of quality of the software.
● CO5: Apply the project management tools, estimation techniques to handle the project.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">UNIT I: Overview of System Analysis & Design, Software Development Life Cycle, Waterfall Model , Spiral Model, Feasibility Analysis, Technical Feasibility, Cost- Benefit Analysis, COCOMO model. [10L]

UNIT II: System Requirement Specification – DFD, Data Dictionary, ER diagram, Process Organization & Interactions. [10L]

UNIT III: System Design – Problem Partitioning, Top-Down And Bottom-Up design; Decision tree, decision table and structured English; Functional vs. Object- Oriented approach. [10L]

UNIT IV: Coding & Documentation - Structured Programming, OO Programming, Information Hiding, Reuse, System Documentation. Testing – Levels of Testing, Organizing for software testing; Software Testing Strategy; Unit Testing: Unit Test Considerations; Integration Testing, OO testing, Reliability Assessment, Validation & Verification Metrics, Monitoring & Control. [8L]

UNIT V: Software Project Management– Project Scheduling, Staffing, Software Configuration Management, Quality Assurance, Project Monitoring. [4L]
CASE TOOLS : Concepts, use and application.</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

1. Roger S. Pressman, Software Engineering: A practitioner's approach, McGraw Hill.
2. Ian Sommerville, Software Engineering, Pearson.</td></tr>
</table>

	Reference Books: 1. Rajib Mall, Fundamentals of Software Engineering, Prentice Hall India. 2. Pankaj Jalote, An integrated approach to Software Engineering, Springer/Narosa.

CSO742 Multimedia Technologies 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 742</td><td>Multimedia Technologies</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Knowledge of data structures databases and compression techniques</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: In depth understanding of media characteristics and resource requirement.
• CO2: Understanding text, graphics. Audio, video media types.
• CO3: Knowledge of issues on dealing simultaneously with multiple data formats, temporal and spatial constraints, synchronization aspects, SAS factors.
• CO4: Understanding of data compression techniques of different media.
• CO5: Understanding of multimedia database storage and retrieval.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Overview of multimedia system: Text, audio, video and graphics. (3L)
Video and Animation: Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options. Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Virtual Reality, Video signal representation, Computer Video Format, Computer- Based animation, Animation Language, Methods of controlling Animation, Display of Animation Transmission of Animation. (10L)
Information representation, media synchronisation, SAS factors, relative and absolute temporal specifications, networking delays, Skew, Jitter. (6L)
Data Compression: Storage Space requirement, Coding Requirements Source, Entropy Coding, Lossy Sequential DCT- based Mode, Expanded Lossy DCT-based Mode, JPEG and MPEG. (8L)
Multimedia file systems: Difference of MM file systems with traditional systems, disk management, disk scheduling, common scheduling algorithms. (5L)
Multimedia databases, multimedia query types, index structures to handle multimedia databases, data storage and retrieval. (10L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books: Multimedia: Computing, Communications and Applications, Ralf Steinmetz. and Klara Nahrstedt, Pearson Education Asia.
Multimedia Communications, Applications, Networks, Protocols and Standards, Fred Halsall, Pearson Education Asia.
Multimedia Systems, John F. Koegel Buford, Pearson Education Asia.

Reference Books: Subrahmanian and Jajodia, Multimedia Database Systems, Springer.</td></tr>
</table>

CSO743 Computer Networks 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 743</td><td>Computer Networks</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Fundamental knowledge in Data Structures</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
● CO2: Comprehend the fundamentals of Physical layer, and will apply them in real time applications.
● CO3: Identify data link layer concepts, design issues, and protocols.
● CO4: Classify the routing protocols and analyze how to assign the IP addresses for the given network.
● CO5: Acquire knowledge of Application layer and Presentation layer paradigms and protocols.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Data communications: components, data representation, direction of data flow; physical structure (type of connection, topology), categories of network (LAN, MAN, WAN); Protocols and standards; Reference models: OSI reference model, TCP/IP reference model. [3L]

Physical Layer: Overview of data (analog & digital), signal (analog & digital), transmission (analog & digital) & transmission media (guided & unguided); Circuit switching: time division & space division switch, TDM bus. [5L]

Data link Layer: Types of errors, error detection & correction methods; framing, Flow control Protocols: Stop & wait ARQ, Go-Back- N ARQ, Selective repeat ARQ, Medium Access sublayer: Token Ring; Reservation, Polling, Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA. [12L]

Network layer: Internetworking & devices, Addressing: IP addressing, subnetting; Routing : techniques, static vs. dynamic routing , Unicast Routing Protocols, Congestion Control and Quality of service (QoS). [12L]

Transport layer: Process to Process delivery; Socket address, UDP; TCP. [5L]

Application Layer: Introduction to DNS, SMTP, SNMP, FTP, HTTP & WWW. [5L]</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

1. B. A. Forouzan – “Data Communications and Networking (3rd Ed.) “ – TMH.
2. A. S. Tanenbaum – “Computer Networks (4th Ed.)” – Pearson Education/PHI.
3. W. Stallings – “Data and Computer Communications (5th Ed.)” – PHI/ Pearson Education.</td></tr>
</table>

	Reference Books: 1. Comer – "Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)" – Pearson Education/PHI.

EIGHTH SEMESTER

Sl. No	Sub. Code	Subject	L-T-S	Credits	Hours
1	CSE810 --	Depth Elective - 6	3-0-0	3	3
2	YYO84*	Open Elective - 4	3-0-0	3	3
3	YYO84*	Open Elective - 5	3-0-0	3	3
4	CSS851	Project - II	0-0-15	5	15
5	CSS852	Project Seminar	0-0-0	1.5	0
6	CSS853	Viva Voce	0-0-0	1	0
		TOTAL	**9-0-15**	**16.5**	**24**

8th Semester

	Basket of Depth Elective – 6
CSE811	Distributed Systems
CSE812	Computer Vision
CSE813	Optical Networks
CSE814	Internet of Things
CSE815	Cloud Computing
CSE816	Mobile Computing
CSE817	Expert Systems

8th Semester

	Basket of Open Elective – 4, 5
CSO851	Machine Learning
CSO852	Data Analytics
CSO853	Distributed Computing
CSO854	Game Theory and its Applications
CSO855	Information Security

CSO856	**Optical Network**
CSO841	**CAD for VLSI**
CSO842	**Internet and Web Technologies**
CSO843	**Soft Computing Techniques**
CSO844	**Compiler Design**

Depth Elective – 6

CSE 811 Distributed Systems 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE811</td><td>Distributed Systems</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Operating systems. Computer Networks</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To explain the paradigm of distributed computing.
● CO2: To explore various existing and possible architectures of distributed systems.
● CO3: To properly appreciate the issues that arise in distributed systems and explore solutions for the problems.
● CO4: To fully appreciate the advantages to be obtained from a distributed environment wrt fault tolerance, load sharing etc.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Distributed Systems. Motivations. Design Issues. (3L)
Clocks in a Distributed System. Synchronization Issues. Logical Clocks. Causal relationships. Vector Clocks. (3L)
Distributed State Detection. Global State. Consistent Cut. Global State recording algorithm. (2L)
Termination Detection. Credit based algorithm. Diffusion Computation based algorithm. (2L)
Distributed Mutual Exclusion. Token based and non-token based algorithms. (4L)
Deadlocks in Distributed Systems. Resource allocation Models. Deadlock Prevention. Deadlock Avoidance – Safe states. Deadlock detection and Correction. Phantom Deadlocks. Centralized, Distributed and Hierarchical deadlock detection algorithms. (5L)
Fault recovery. Classes of Faults. Backward and Forward recovery. Log based recovery. Checkpoints. Shadow paging. (5L)
Fault Tolerance. Data Replication. Quorum Algorithms. Distributed Commit Protocols. 2-phase commit. 3-phase commit. Election Algorithms. Bully algorithm. Ring topology algorithm. (8L)
Byzantine faults and Agreement Protocols. (2L)
Distributed File systems. Mechanisms. Stateful and Stateless servers. Scalability. Naming and Name Servers. (4L)
Distributed Scheduling. Load Balancing. Load Estimation. Stability. Process Migration. Remote Procedure Calls. Transparency. Binding. (4L)</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. Advanced Concepts in Operating Systems. Singhal and Sivaratri. McGraw Hill. **Refenence Books:** 1. Operating Systems: A Concept Based Approach. Dhamdhere. McGraw Hill. 2. Distributed Operating Systems: Concepts and Design. P.K.Sinha. Prentice Hall. 3. Distributed Operating Systems. A.Tanenbaum. Pearson Education. 4. Distributed Systems: Concepts and Design. Coulouris et.al. Pearson Education.

CSE 812 Computer Vision 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 812</td><td>Computer Vision</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Probability and Statistics, Algebra, Optimization, Computer Graphics</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding basic architecture and principles of computer vision systems.
● CO2: Implementation of computer vision algorithms including depth estimation, multi-camera view and motion analysis components.
● CO3: Apply basic image processing and feature extraction techniques in order to design computer vision algorithms.
● CO4: Analysis of pattern analysis and image segmentation techniques used for computer vision systems.
● CO5: Design and development of real time computer vision systems.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Digital Image Formation and low-level processing: Overview and State-of-the-art, Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc; Fourier Transform, Convolution and Filtering, Image Enhancement, Restoration, Histogram Processing. [5L]

Depth estimation and Multi-camera views: Perspective, Binocular Stereopsis: Camera and Epipolar Geometry; Homography, Rectification, DLT, RANSAC, 3-D reconstruction framework; Auto-calibration. Apparel. [6L]

Feature Extraction: Edges - Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT. [8L]

Image Segmentation: Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection. [5L]

Pattern Analysis: Clustering: K-Means, K-Medoids, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised; Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA; Non-parametric methods. [8L]

Motion Analysis: Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo; Motion parameter estimation. [4L]

Shape from X: Light at Surfaces; Phong Model; Reflectance Map; Albedo estimation; Photometric Stereo; Use of Surface Smoothness Constraint; Shape from Texture, color, motion and edges. [6L]</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** ● Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011. ● D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Pearson Education, 2003. **Reference Books:** ● Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004. ● K. Fukunaga; Introduction to Statistical Pattern Recognition, Second Edition, Academic Press, Morgan Kaufmann, 1990. ● R.C. Gonzalez and R.E. Woods, Digital Image Processing, Addison- Wesley, 1992. **Others:** Swayam Online Course 1. https://swayam.gov.in/nd1_noc19_cs58/preview 2. https://www.coursera.org/courses?query=computer%20vision 3. https://www.edx.org/course/computer-vision-and-image-analysis-3 4. https://www.mooc-list.com/tags/computer-vision

CSE 813 Optical Networks 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 813</td><td>Optical Networks</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Basic Concepts of Computer Networks, and Algorithms</td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Identify and illustrate the main differences between optical networking and traditional networking, different problems in optical networks.
● CO2: Comprehend the routing and wavelength assignment (RWA), virtual topology design, wavelength rerouting, Traffic grooming in WDM optical network design.
● CO3: Understanding of the wavelength convertible network.
● CO4: Concept and analyze the benefit of various survivability strategies.
● CO5: Comprehend the multicast routing in optical networks.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Fundamentals and Different Problems: Optical fiber principles, Optical transmission system, Wavelength Division Multiplexing(WDM), optical networking evolution, Optical Network Architectures, Different issues in wavelength routed networks.
(06L)
2. Routing and Wavelength Assignment (RWA) algorithms: ILP formulation of the RWA problem, Route Selection algorithms – Fixed Routing, Fixed Alternate Routing, Exhaust Routing, Least Congested Path Routing, Limited alternate Routing. Wavelength Selection algorithms. Joint wavelength-Route selection algorithm.
(07L)
3. Wavelength Convertible Networks: Need for Wavelength Converters, Wavelength convertible Switch Architecture, Routing in Convertible Networks, Performance Evaluation of Convertible networks, Network with Sparse Wavelength Conversion, Converter Placement problem.
(06L)
4. Wavelength Rerouting Algorithm: Benefits of wavelength rerouting, Issues in wavelength rerouting, Rerouting algorithm.
(04L)
5. Virtual Topology Design: Physical and Virtual topology, Traffic routing over virtual topology, Limitations on virtual topology, Virtual topology problem formulation, Virtual topology design heuristics.
(06L)</td></tr>
</table>

	6. **Traffic Grooming:** Basic concepts, Grooming node architecture, ILP formulation of the traffic grooming problem, Different heuristics (MST, MRU, TGCP, etc) for the traffic grooming problem. (06L) 7. **Optical Multicast Routing**: Multicast routing problem, architecture of Light spliting node and MI node, Network with full splitting and sparse splitting, Multicast Tree generation algorithms – Source based, Steiner based and Virtual source based tree generation algorithms. (07L)
Text Books, and/or reference material	**Text Books:** **1.** WDM OPTICAL NETWORKS Concepts, Design and algorithms. by C. Siva Ram Murthy and Mohan Gurusamy (PHI). 2. OPTICAL NETWORKS by Biswanath Mukherjee (TMH). **Reference Books:** 1. Optical Networks: A Practical Perspective (3rd Edition) by R. Ramaswami, K. Sivarajan, G. Sasaki (Morgan Kaufmann Publishers).

CSE 814 Internet of Things 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 814</td><td>Internet of Things</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understand the concepts of Internet of Things.
● CO2: Preparing the right background to take up research works in emerging wireless technologies and Internet of Things.
● CO3: To introduce the scopes of using sensing, edge computing, Machine learning mechanisms in pervasive cyber physical systems.
● CO4: Able to understand the innovation opportunity in IoT application segments.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Module 1: Introduction to IoT and Sensing (3L)
Introduction to IoT, Sensing, Edge computing, Data processing, Learning.

Module 2: Sensing Layer (4L)

Different type of sensors, working principle of some sensors like Ultrasonic sensor, Thermal Sensors, Infrared Sensors, Pollutant Sensors, Temp, IMU Sensor etc.

Module 3: Play with Sensors & Microcontroller/Microcomputer (4L)

Open source hardware, Play with Sensors using Arduino Programming, Local data processing using Raspberry Pi/Uddo Neo, Play with different Network Modules (Bluetooth, WiFi, GSM/GPRS).

Module 4: Wireless Networks Present and Future (10L)

Concept of TCP/IP protocol Stack, 802.11 Protocol (WiFi Network), LoRa Network, Acoustic Communication, Socket Programming, Wireshark Tool

Module 5: IoT Protocols (4L)

HTTP, QUIC, CoAP, MQTT.

Module 6: Performance and Security in IoT (4L)

Module 7: Case Study of IoT Based Applications (14L)</td></tr>
</table>

	Case Study 1: **(activity Identification)** Human Activity using Ultra sonic Sensors/Thermal Sensors. Case Study 2: **(Environment Monitoring)** Pollution Monitoring and Forecasting in Indoor and Outdoor. Case Study 3: **(Road Transportation System)** (a)Important PoIs using GPS trails, (b)Context Aware Speed Profiling from Mobile Phone Sensors, (c)My Smartphone Can Monitor My Street-lights. Case Study 4: **(Challenged Networks)** offline Crisis Mapper Design using ChatBot, IoT Protocol Stack Development using Acoustic Communication. Case Study 5: **(Agriculture Monitoring):** Smart Farming using MQTT Protocol through Cost-effective Heterogeneous Sensors.
Text Books, and/or reference material	**Text Books** **1.** "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press). 2. "Internet of Things: A Hands-on Approach", by Arshdeep Bahga and Vijay Madisetti (Universities Press).

CSE 815 Cloud Computing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 815</td><td>Cloud Computing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Explain the core concepts of the cloud computing paradigm: paradigm shift, the characteristics, advantages and challenges various models and services.
● CO2: Apply fundamental concepts in cloud infrastructures to understand the tradeoffs in power, efficiency and cost, and then study how to leverage and manage single and multiple datacenters to build and deploy cloud applications that are resilient, elastic and cost-efficient.
● CO3: Learning of system, network and storage virtualization and outline their role in enabling the cloud computing system model.
● CO4: Analyze the performance, scalability, and availability of the underlying cloud technologies and software.
● CO5: Identify security and privacy issues in cloud computing.
● CO6: Explain recent research results in cloud computing and identify their pros and cons.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Services Oriented Computing - Service Oriented Software, Web Applications Paradigm.[2]

Services Oriented Architecture - SOA and Web Services Fundamentals, SOA and Service-Orientation, SOA - Planning and Analysis, SOA - Technology and Design, SOA Reference model (OASIS), SOA standard S3, Business Process and SOA, Software as a Service (SaaS) [4]

Web Services - Introduction to Web Services, Web Service Jargon – Publishing, Discovery and Binding, Web Service Technologies – WSDL, SOAP, UDDI, Issues and Challenges – MANET, CLOUD, DTN,Formal, Representation of Services[4]

Cloud Computing Basics- Overview, Applications, Intranets and the Cloud. Organization and Cloud Computing- Benefits, Limitations, Security Concerns. [2]

Cloud Infrastructure - Data center, Virtualization, Clients, Security, Network, Services and Delivery Models (SaaS, PaaS, IaaS). Case study like Amazon EC2, Microsoft Azure etc. Deployment types (Private, Public, Hybrid) [4]

Software as a Service (Saas)- Understanding the Multitenant Nature of SaaS Solutions, Understanding SOA. [2]

Platform as a Service (PaaS)- IT Evolution Leading to the Cloud, Benefits of Paas Solutions, Disadvantages of Paas Solutions. [2]</td></tr>
</table>

	Infrastructure as a Service (Iaas)-Understanding IaaS, Improving Performance through Load Balancing, System and Storage Redundancy, Utilizing Cloud-Based NAS Devices, Advantages, Server Types. [3] **Virtualization-**Understanding Virtualization, History, Server Virtualization, Data Storage Virtualization. [4] **Securing the Cloud**- General Security Advantages of Cloud-Based Solutions, Introducing Business Continuity and Disaster Recovery. Disaster Recovery- Understanding the Threats. [4] **Migrating to the Cloud-**Cloud Services for Individuals, Cloud Services Aimed at the Mid-Market, Enterprise-Class Cloud Offerings, and Migration. [4] **Designing Cloud Based Solutions-**System Requirements, Design Is a Give-and-Take Process. Coding Cloud Based Applications-Creating a Simple Yahoo Pipe, Using Google App Engine and creating a Windows Azure Application. Application Scalability-Load-Balancing Process, Designing for Scalability, Capacity Planning Versus Scalability, Scalability and Diminishing Returns and Performance Tuning. [8]
Text Books, and/or reference material	**Text Books:** Cloud Computing: A Practical Approach by Anthony T. Velte Toby J. Velte, Robert Elsenpeter, The McGraw-Hill Publisher. Cloud Computing: SaaS, PaaS, IaaS, Virtualization and more. by Dr. Kris Jamsa, Jones & Bartlett Publisher. **Reference Books:** Cloud Computing Bible by Barrie Sosinsky, Published by Wiley Publishing. Cloud Computing for Dummies by Judith Hurwitz, Robin Bloor, Marcia Kaufman, and Dr. Fern Halper, Wiley Publishing. Cloud Computing Theory And Practice Danc.Marinercus, Elsevier.

CSE 816 Mobile Computing 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 816</td><td>Mobile Computing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Computer Networks</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Introduce to the basic of of Wireless Networks.
● CO2: Preparing the right background to take up research works in emerging wireless technologies and Internet of Things.
● CO3: To introduce the scopes of using sensing, edge computing, Machine learning mechanisms in pervasive cyber physical systems.
● CO4: Able to understand the innovation opportunity in IoT application segments.
● CO5: Hands-on experience on Wireless Networks & Mobile Computing.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Module 1: Physical Layer (6 Hours)
Bit transmission over Wireless, Vary Much different from Wired Network.

Module 2: Mac Layer (8 Hours)
Access in Shared Medium, Difference between Wired MAC & Wireless MAC, Different Type of MACs (a) Random MAC (b) Scheduled MAC, Examples of MAC Implementation (WiFi Protocol --802.11, Bluetooth Protocol--805.15).

Module 3: Network Layer (8 Hours)
Reactive Routing, Proactive Routing, DSR Principle, AODV Principle, Location Aware Routing. Adhoc Network, Delay Tolerant Network, Opportunistic Network Introduction, Architecture & Applications, Routing Algorithms – Epidemic, Prophet, Spray & Wait, Spray & Focus, Maxprop Simulation Tool - ONE Simulator.

Module 4: Transport Layer (8 Hours)
Wireless TCP and rationale, Difference between Wired TCP and Wireless TCP, QoS Measurement of Wireless Networks.

Module 5: Modelling (8 Hours)
Mathematical Modelling of Network Functionalities - Combining them to derived overall performance.

Module 6: Case Study: Implementation of opportunistic Networks in Challenged Network scenarios (4 hours)
(a) Connection Mechanism (b) Sync - Transferring the information in Collaborative manner (c) Offline Dashboard (Information Summarization) (d)security</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. “Mobile Communication”, by Jochen Schiller (PEARSON EDUCATION LIMITED).
2. “Wireless Networking” A kumar, D. manjunath, J. Kuri, Elsevier, 2008.
3. “Wireless Communication”, T. S. Rappaport, Pearson, latest edition.</td></tr>
</table>

	Reference Books: **Research Papers:** 1. IEEE Infocom Tutorials slides by Prof. Nitin Vaidya. **Others:** Tools: ● Sniffer Tool (Wireshark) ● Simulation Tools: OMNET ONE NS3

CSE 817 Expert Systems 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSE 817</td><td>EXPERT SYSTEMS</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites
Artificial Intelligence, Data Mining, Pattern Recognition, OOP</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3"></td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Idea about Knowledge Base & Expert Systems.
● CO2: Idea of Inference Tool and Inference Engine and different methods of Inference Methodologies.
● CO3: Idea about Reasoning under Uncertainty and Uncertainty Management which is really crucial under present day scenario.
● CO4: Concept of the Design of Expert System Components and Experts Systems.
● CO5: Some Examples of Practical Experts System.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Introduction to Expert Systems: Definition of an Expert System – Advantages of Expert Systems – Characteristics of Expert Systems - Applications and Domains – Procedural and Non procedural Systems. [6L]

2. The Different Techniques for Knowledge Representation: Meaning of Knowledge – Productions – Semantic Nets- Frames – Logics – Propositional and Predicate Logic – The universal and existential quantifiers. [7L]

3. The Different Methods of Inference : Trees, Lattice and Graph – State and Problem Space – Rules of Inference – Logic Systems – Resolution Systems and Deductions – Forward and Backward Reasoning – Meta knowledge. [7L]

4. The Reasoning Under Uncertainty and Inexact Reasoning – Uncertainty – Types of Errors – Classical Probability – Experimental and Subjective probabilities – Compound and Conditional Probabilities – Temporal Reasoning – Uncertainty in Inference Chains – Evidence Combination – Uncertainty and Rules – Certainty Factors – Dempster- Shafer Theory – Approximate Reasoning. [8L]

5. The Design of Expert Systems Tool and Expert Systems : Selecting Appropriate Problem – Stages in the development – Errors in Development – Expert System Life Cycle – A Life Cycle Model. [7L]

6. Some Practical Examples of Expert System Design – Modular Design – Phases and Control Facts – Importing and Exporting facts – Modules and Execution Control – Certainty Factors – Decision Trees – Backward Chaining – A Monitoring Problem.[7L]</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. Expert Systems Principles and Programming – Bikash Publishing House. 2. Pattern Classification- – Duda, Hart & Stork – J. Wiley & Sons. **Reference Books:** 1. Artificial Neural Networks – B. Yegnanarayana – PHI. 2. Neural Networks for Pattern Recognition – C.M. Bishop – Oxford.

Open Elective – 4, 5

CSO 851 Machine Learning 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 851</td><td>Machine Learning</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Basic concept of Probability and Statistics.</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Understanding of the basic concepts, fundamental issues and challenges of machine learning.
● CO2: Comprehend the principle and techniques of supervised learning.
● CO3: Explain the basic concepts and techniques of unsupervised learning.
● CO4: Understanding of the basic concepts and challenges of reinforced learning.
● CO5: Ability to apply the concepts of machine learning in different domains.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Introduction: what is Machine Learning; Human learning and Machine learning; Well-posed learning problem; Types of Machine Learning: Supervised, Unsupervised, and Reinforcement learning; Applications, Issues, and tools of Machine Learning. (03 L)
2. Concept Learning: Inductive learning hypothesis, general to specific ordering of hypothesis; FIND-S algorithm; Version space, candidate elimination algorithm; Inductive bias. (04 L)
3. Bayesian Learning, Naïve Bayes Classifier, Optimal Classifier. (03 L)
4. Supervised learning: Classification- k-Nearest Neighbour, Decision Tree, Support vector machine. Regression- Simple and Multiple linear regression. (12L)
5. Artificial Neural Networks: Biological neuron and artificial neuron, How ANN works, Parallel distributed model of ANN; Activation functions; Perceptron, McCulloch-Pits model, ADALINE network model; Architecture of ANN- single-layer feed forward, multi-layer feed forward, competitive network, recurrent network; Backpropagation algorithm; Basic concept of deep learning. (05L)
6. Unsupervised learning: Different clustering techniques- Partitioning methods (k-means, k-medoid, etc. clustering techniques), Hierarchical methods (Agglomerative and Divisive techniques: MIN, MAX, Group average, Ward's etc. methods), and Density-based method (DBSCAN). (05 L)
7. Unsupervised learning: Rule mining and Association analysis- different terminology (itemset, support count, support, association rule, confidence, etc.); Association rule</td></tr>
</table>

	mining techniques; Market-Basket analysis; Apriori principle, Apriori algorithm for frequent itemset generation, Rule generation for apriori algorithm. (05 L) 8. Genetic Algorithm based Learning. (02 L) 9. Reinforcement Learning: Basic concept, Model based learning, Temporal difference based learning. (03 L)
Text Books, and/or reference material	Text Books: 1. Machine Learning by Tom Mitchell [Mc. Graw-Hill]. 2. Machine Learning by S. Dutt, S. Chandramouli, and /A. K. Das [Pearson, 2019]. 3. Applied machine Learning by M. Gopal [Mc. Graw-Hill, 2018]. 4. NPTEL Course materials. Reference Books: Introduction to Machine Learning by Ethem Alpaydin [MIT Press].

CSO 852 Data Analytics 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 852</td><td>Data Analytics</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Knowledge in handling and analyzing extremely large datasets.
● CO2: Learns the techniques of uncovering hidden patterns, correlations and other insights out of these datasets.
● CO3: Ability to apply the concepts of data analytics in different domains.
● CO4: Ability to contextually integrate and correlate large amounts of information.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Data Analytics, Types of Data Analytics: Descriptive Analytics, Diagnostic Analytics, Predictive Analytics, and Prescriptive Analytics. Use Cases, Issues and Challenges in Big Data Analytics. (4L)
Fundamentals of Statistics: Population, Sample, Parameter, Statistic, Variable. Descriptive Statistics, Inferential Statistics. Basic Probability Theory: Random Experiment, Sample Space, Random Variables, Probability, Conditional Probability, Independence, Conditional Independence, Expectation, Variance, Probability Distribution, Joint Probability Distribution, Conditional Probability Distribution. (8L)
Similarity Measures: Jaccard Similarity, Cosine Similarity, Adjusted Cosine Similarity. Missing Value Prediction Techniques: Mean Centering, Weighted Average, Z-Score. (6L)
Basics of Complex Network: Scale-Free Networks, Small-World Phenomenon, Degree Distributions, Transitivity or Clustering. Centrality Measures: Degree Centrality, Betweenness Centrality, Closeness Centrality, Eigenvector Centrality, PageRank Centrality. Community Structure, Community Detection Algorithms: Girvan-Newman, Fast Greedy, Label Propagation, Clique Percolation Method. Community Quality Metrics: Modularity, NMI, Conductance. (10L)
Introduction to Data Mining, Machine Learning Techniques: Least Square Regression, Decision-trees, SVM. Clustering Techniques: K-Means. (8L)
Introduction to Hadoop Ecosystem – HDFS, Map-Reduce, PIG, HIVE, HBase, Mahout, Zookeeper, Flume, Sqoop, etc. (6L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
3. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data – EMC Education Services – Wiley.
4. Machine Learning: Hands-On for Developers and Technical Professionals – Jason Bell – Wiley.
Reference Books:
3. Networks: An Introduction – M. E. J. Newman – Oxford University Press.
4. Hadoop: The Definitive Guide – Tom White – O'Reilly.</td></tr>
</table>

CSO 853 Distributed Computing 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO853</td><td>Distributed Computing</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Operating Systems, Computer networks.</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To explain the paradigm of distributed computing.
● CO2: To explore various existing and possible architectures of distributed systems.
● CO3: To properly appreciate the issues that arise in distributed systems and explore solutions for the problems.
● CO4: To fully appreciate the advantages to be obtained from a distributed environment wrt fault tolerance, load sharing etc.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction to Distributed Systems. Motivations. Design Issues. (3L)
Clocks in a Distributed System. Synchronization Issues. Logical Clocks. Causal relationships. Vector Clocks. (3L)
Distributed State Detection. Global State. Consistent Cut. Global State recording algorithm. (2L)
Termination Detection. Credit based algorithm. Diffusion Computation based algorithm. (2L)
Distributed Mutual Exclusion. Token based and non-token based algorithms. (4L)
Deadlocks in Distributed Systems. Resource allocation Models. Deadlock Prevention. Deadlock Avoidance – Safe states. Deadlock detection and Correction. Phantom Deadlocks. Centralized, Distributed and Hierarchical deadlock detection algorithms (5L)
Fault recovery. Classes of Faults. Backward and Forward recovery. Log based recovery. Checkpoints. Shadow paging. (5L)
Fault Tolerance. Data Replication. Quorum Algorithms . Distributed Commit Protocols. 2-phase commit. 3-phase commit. Election Algorithms. Bully algorithm. Ring topology algorithm. (8L)
Byzantine faults and Agreement Protocols. (2L)
Distributed File systems. Mechanisms. Stateful and Stateless servers. Scalability. Naming and Name Servers. (4L)
Distributed Scheduling. Load Balancing. Load Estimation. Stability. Process Migration. Remote Procedure Calls. Transparency. Binding. (4L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:

Advanced Concepts in Operating Systems. Singhal and Sivaratri. McGraw Hill.</td></tr>
</table>

	Reference Books: 1. Operating Systems : A Concept Based Approach. Dhamdhere. McGraw Hill. 2. Distributed Operating Systems : Concepts and Design. P.K.Sinha. Prentice Hall. 3. Distributed Operating Systems. A.Tanenbaum. Pearson Education. 4. Distributed Systems : Concepts and Design. Coulouris et.al. Pearson Education.

CSO 854 Game Theory and its Applications 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 854</td><td>Game Theory and its Applications</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">1. MAC 01: Mathematics - I
2. MAC 02: Mathematics - II</td><td colspan="6">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2"></td><td colspan="6">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Can have the efficiency to act in a strategic situation.
● CO2: Can analyse the strategic interactions among agents.
● CO3: Can understand modern state of the art in Game Theory.
● CO4: Will have the knowledge of related area where Game Theory can be applied.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Introduction: Motivation to the course. (2L)
Non-Coperative Game Theory: Introduction to Game Theory, Extensive Form Games, Strategic Form Games, Dominat Strategy Equilibria, Pure Strategy Nash Equlibrium, Mixed Strategy Nash Equilibrium with examples. (8L)
Mechanism Design without Money: One sided and two sided matching with strict preferences, Voting theory, and Participatory democracy. (5L)
Mechanism Design with Money: Auction basics, sponsored search auctions, Revenue optimal auctions, VCG Mechanisms. Online auctions. (6L)
Cooperative Game Theory: Coalitional Games, The Core, and The Shapley Value. (4L)
Repeated Games: Introduction to repeated games and its Applications. (4L)
Applications: Incentive Study in - P2P Networks, Crowdsourcing. (5L)
Some Special Topics: Fair Division, Price of Anarchy, Scoring rules, Learning in Auction, Synergies between Machine Learning & Game Theory. (8L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. Algorithmic Game Theory. Cambridge University Press, New York, NY, USA, 2007, ISSN: 978-0521872829.
2. M. Maschler, E. Solan, and S. Zamir. Game Theory, Cambridge University Press; 1st Edition, ISSN: 978-1107005488, 2013.
3. Y. Narahari. Game Theory and Mechanism Design. World Scientific Publishing Company Pte. Limited, 2014, ISSN: 978-9814525046.
4. T. Roughgarden, Twenty Lectures on Algorithmic Game Theory, Cambridge University Press, 2016, ISSN: 978-1316624791.</td></tr>
</table>

	Reference Books: 1. T. Roughgarden, CS364A: Algorithmic Game Theory Course (Stanford University), 2013. 2. T. Roughgarden, CS269I: Incentives in Computer Science Course (Stanford University), 2016. 3. S. Barman and Y. Narahari, E1:254 Game Theory Course (IISc Bangalore), 2012.

CSO 855 Information Security 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="7">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 855</td><td>Information Security</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="2">Programming Languages, Computer Networks and Operating Systems</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">By the end of the course, Students will be able to:
● CO1: Learn fundamental concepts of Information Security viz. Security Models (like the CIA triad), Access Control Mechanisms, Security policies and Security Mechanisms like authentication, identification, authorization, non-repudiation, etc.
● CO2: Understand program security issues, attack vectors, and malicious code including worms, viruses, and Trojan horse and logic bombs.
● CO3: Understand common vulnerabilities like Buffer Overflow, TOC-TOU flaws. Learn secure programming requirements; write robust security code and exploit/recreate-and-fix common vulnerabilities in software.
● CO4: Define trusted computing base and Operating System hardening as defence mechanisms, Intrusion detection and prevention.
● CO5: Get introduced to trusted computing and multilevel security.
● CO6: Explain concepts related to applied cryptography, including plain-text, cipher-text, four techniques for crypto-analysis, symmetric cryptography, asymmetric cryptography, digital signature, message authentication code, hash functions, and modes of encryption operations.
● CO7: Explain and compare security mechanisms for conventional operating systems, OS hardening. Case Study on Linux.
● CO8: Exposed to network and distributed systems security issues and solutions including authentication, key distribution and management and network security protocols like SSL/TLS.
● CO9: Introduced to Laws and regulatory requirements, security standards and controls, risk management, security metrics and performance indicators, security auditing, education, training and awareness and digital forensics.</td></tr>
<tr><td>Topics Covered</td><td colspan="6">▪ Information Security Introduction -- Defining and Understanding security through security models, Confidentiality, Integrity and Availability, formal description of</td></tr>
</table>

	security, Attacks and Defences, Threats, Vulnerabilities and Risk, Assurance, Prevention, Detection, Security Controls. [2L] ▪ Identification and Authentication. [2L] ▪ Authorization and Access Control, Access Control Models & Mechanisms and Multilevel Security. [2L] ▪ Auditing and Accountability. [2L] ▪ Computational Number Theory & Cryptography -- Fermat's theorem, Euler's theorem, Euclid's algorithm, manually and computationally encrypt/decrypt, sign/verify signatures for small messages using RSA, Deffie-Hellman and DSA algorithms. Applied cryptography viz. Symmetric key Cryptography, asymmetric Cryptography and Digital Signatures, message authentication codes, hash functions and modes of cryptographic operations.[6L] ▪ Physical Security. [1L] ▪ Network Security – Network threats: eavesdropping, spoofing, modification, denial of service attacks o Introduction to network security techniques: firewalls, virtual private networks, intrusion detection. Different Network Security Protocols.[6L] ▪ Operating System Security & Trusted OS-- Memory, time, file, object protection requirements and techniques, Protection in contemporary operating systems, ACLs, DAC, MAC, RBAC, Identification and authentication, Identification goals, Authentication requirements, Human authentication, Machine authentication, OS Forensics. Assurance & Trust, Design principles, Evaluation criteria, Evaluation process.[8L] ▪ Application & Program Security– Flaws, Malicious code: viruses, Trojan horses, worms, Program flaws: buffer overflows, time-of-check to time-of-use flaws, incomplete mediation o Defenses, Software development controls, Testing techniques.[5L] ▪ Secure Coding. [2L] ▪ Distributed Systems Security. [2L] ▪ Digital Forensics. [2L] ▪ Cyber Laws. [2L]
Text Books, and/or reference material	**Text Books:** 1. The Basics of Information Security by Jason Andress, Syngress Publication. 2. Security in Computing (3rd Edition) 3rd Edition by Charles P. Pfleeger (Author), Shari Lawrence Pfleeger (Author), PHI. 3. B. Tjaden Fundamentals of Secure Computer Systems Franklin Beedle & Associates 2003. 4. D. Russell & G.T. Gangemi, Sr, Computer Security Basics. 5. W. Stallings, Network Security Essentials. Prentice Hall, 2003.

CSO 856 Optical Network 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 856</td><td>Optical Network</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Basic Concepts of Computer Networks</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Identify the different problems in optical networks.
● CO2: Understanding the routing and wavelength assignment (RWA), virtual topology design, wavelength rerouting, Traffic grooming in WDM optical network design.
● CO3: Understanding of the wavelength convertible network.
● CO4: Comprehend the multicast routing in optical networks.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">1. Fundamentals and Optical Components: Optical fiber principles, Optical transmission system, Wavelength Division Multiplexing(WDM), optical networking evolution, Optical Network Architectures; Optical Components- Couplers, Multiplexers and Filters, Optical Amplifiers, Transmitter, Detectors, switches and wavelength converters; Different issues in wavelength routed networks. (12L)
2. Routing and Wavelength Assignment (RWA) algorithms: ILP formulation of the RWA problem, Route Selection algorithms – Fixed Routing, Fixed Alternate Routing, Exhaust Routing, Least Congested Path Routing, Limited alternate Routing. Wavelength Selection algorithms. Joint wavelength-Route selection algorithm. (08L)
3. Wavelength Convertible Networks: Need for Wavelength Converters, Wavelength convertible Switch Architecture, Routing in Convertible Networks, Performance Evaluation of Convertible networks, Network with Sparse Wavelength Conversion, Converter Placement problem. (04L)
4. Wavelength Rerouting Algorithm: Benefits of wavelength rerouting, Issues in wavelength rerouting, Different rerouting algorithms. (05L)
5. Virtual Topology Design: Concept of virtual topology, Limitations on virtual topology, Virtual topology problem formulation, Virtual topology design algorithms. (06L)
6. Traffic Grooming: Basic concepts, Grooming node architecture, ILP formulation of the traffic grooming problem, Different heuristics (MST, MRU, TGCP, etc) for the traffic grooming problem. (05L)
7. Basic concepts of Multicast routing and wavelength assignment. (02L)</td></tr>
</table>

Text Books, and/or reference material	**Text Books:** 1. WDM OPTICAL NETWORKS Concepts, Design and algorithm by C. Siva Ram Murthy and Mohan Gurusamy (PHI). 2. OPTICAL NETWORKS by Biswanath Mukherjee (TMH). **Reference Books:** 1. Optical Networks: A Practical Perspective (3rd Edition) by R. Ramaswami, K. Sivarajan, G. Sasaki (Morgan Kaufmann Publishers).

CSO841 CAD for VLSI 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO841</td><td>CAD for VLSI</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Digital Electronics, Computer Organisation, Algorithm Analysis and Design.</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To visit the various stages of the VLSI design cycle and appreciate the role of automation therein.
● CO2: To appreciate how High Level Synthesis converts an HDL code into an architecture level design.
● CO3: To discuss the algorithmic approach to physical design.
● CO4: To emphasize the importance to testability measures in the design.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">VLSI Design cycle. Design styles. System packaging styles. Fabrication of VLSI devices. Design rules-overview. (3L)
HLS: Scheduling in High Level Synthesis. ASAP and ALAP schedules. Time constrained and Resource constrained scheduling. (4L)
HLS: Allocation and Binding. Datapath Architectures and Allocation tasks. (4L)
Partitioning. Clustering techniques. Group Migration algorithms. (4L)
Floorplanning. Constraint based Floorplanning. Rectangular Dualization. Hierarchical Tree based methods. Simulated Evolution approaches. Timing Driven floorplanning. (5L)
Placement.Simulation based placement algorithms. Partitioning based placement algorithms.ClusterGrowth.(5L)
Global Routing. Maze Routing algorithms. Line probe algorithms. Shortest Path based algorithms. Steiner’s Tree based algorithms. (5L)
Detailed Routing. Channel Routing Algorithms. Switchbox Routing. Over-the-cell routing. Clock and Power Routing. (4L)
Design for testability. Fault testing. Ad-hoc and structured DFT techniques. (8L)</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books:
1. Algorithms for VLSI Physical Design Automation. N.A.Sherwani. Kluwer Academic Publishers.
2. High-Level Synthesis: Introduction to Chip and System Design. Gajski et. al. . Kluwer Academic Publishers.
3. Digital Systems Testing and Testable Design. Abramovici et.al. Jaico Publications.

Reference Books
1. VLSI Physical Design Automation. Sadiq M. Sait and Habib Youssef. Kluwer Academic Publishers.</td></tr>
</table>

	2. Algorithms for VLSI Design Automation. Sabih H. Gerez. Wiley India. 3. Essentials of Electronic Testing for Digital, Memory and Mixed Signal VLSI Circuits. Bushnell and Agrawal. Kluwer Academic Publishers.

CSO842 Internet and Web Technologies 3-0-0 3Credits 3Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO 842</td><td>Internet and Web Technologies</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="2">Pre-requisites</td><td colspan="6">Course Assessment methods (Continuous (CT) and end assessment (EA))</td></tr>
<tr><td colspan="2">Programming Fundamentals, Data Structure and Algorithms, Operating Systems, Data networks (may be carried out simultaneously)</td><td colspan="6">CT+EA [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">• CO1: Understanding the fundamental concepts of Internet Structure and Protocols.
• CO2: Using TCP/IP protocols and Internet programming using SOCKET API.
• CO3: Understanding HTTP protocol and Structures of Web Programming.
• CO4: Designing and developing Web applications with security enhancement.
• CO5: Understanding Semantic Web and Applying Web Analytics over Semantic Web.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">INTERNET TECHNOLOGY:
Brief review of Data Networking; Introduction to Data Communication, OSI Layered Architecture, Introduction to Networking Devices, Network Performance Metrics. (4L)
data transmission over point to point links, link sharing and MACs, Forwarding and Routing, TCP-IP layered network concepts. (3L)
Internet specific issues like scalability, inter operability. (1L)
Internet Structures – logical and physical grouping with sub netting and super netting. (3L)
Review of TCP-IP protocols – processing, performance and variations. (3L)
Security Implementations - secured IP, Transport Layer security. (3L)
Quality of Service Issues and their Application in Internet. (2L)
HTTP: Requests and Responses - Message Formats, Headers and Fields; TCP Keep-alive and pipe-lining concepts; Server Architecture, Performance and Deployment. (3L)
WEB PROGRAMMING: Document Object Model; Client side scripting fundamentals: Server Side Scripting and Programming – Data base connectivity, session management and security enhancement; Introduction to Web Application Development Platforms – JavaEE, Dzango. (7L)</td></tr>
</table>

	XML: DTD and Schema; Visualisation using XSLT; Web Application using XML; Service Oriented Architecture and Web services based application development and deployment; Xquery and SOA based application development platforms. (6L) SEMANTIC WEB: General Concept of Semantic Web and linked Data; RDF based relation description; Web Ontology concepts and use; Putting XML, RDF and Ontology together to develop semantic web applications; Capturing Information from semantic web pages; Data analytics over semantic and linked Web. (7L)
Text Books, and/or reference material	**Text Books:** 1. B. A. Forouzan, "TCP/IP Protocol Suite", 4th Edition, 2010, McGrawHIll Publishers. 2. P. Deitel, H. Deitel, A Deitel, "Internet and World Wide Web – How to Program", Pearson. 3. G. Antoniou, P. Groth, F. Harmelen and R. Hoekstra, "A Semantic Web Primer" Prentice Hall India. **Reference Books:** 1. D. E. Comer and D L Stevens, "Internetworking with TCP/IP vol.II", Pearson. 2. www.w3schools.com

CSO843 Soft Computing Techniques 3-0-0 3 Credits 3 Hours

<table>
<tr><td colspan="8">Department of Computer Science and Engineering</td></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO843</td><td>Soft Computing Techniques</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">Introduction to computing, Data Structures and Analysis of Algorithms</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: To familiarize with neural networks and learning methods for neural networks.
● CO2: To introduce basics of genetic algorithms and their applications in optimization and planning.
● CO3: To introduce the ideas of fuzzy sets, fuzzy logic and fuzzy inference system.
● CO4: To introduce students' tools and techniques of Soft Computing.
● CO5: To develop skills thorough understanding of the theoretical and practical
● aspects of Soft Computing.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">Module I: Introduction (6L)

Introduction and different definitions of Soft Computing with their application in real life problems, Basic tools/members of Soft Computing: Fuzzy Logic, Neural Network and Evolutionary Computing.

Module II: Fuzzy Logic (12L)

Fuzzy Logic-I: Crisp Sets, Fuzzy sets, Fuzzy membership functions, Basic operations on fuzzy sets, Fuzzy relations and Composition of fuzzy relations. Fuzzy Logic –II (Fuzzy Rules and Approximate Reasoning): Fuzzy if-then rules: M-A and TSK Rules, Fuzzification, Compositional rule of Inference/Approximate Reasoning, Defuzzification and Applications.

Module III: Neural Networks (10L)

Neural Networks-1 (Introduction & Architecture): Introduction to neural networks: Artificial Neuron and its model, Activation functions, Neural network architecture, learning algorithms/rules, Training and testing.

Neural Networks-II: Perceptron model: single layer and multilayer perceptron (MLP), Error back propagation, Radial basis function network (RBFN), Self-organizing map network (SOMN).</td></tr>
</table>

	Module IV: Evolutionary Computing (14L**)** **Evolutionary Computing-I:** Evolutionary Computing, Basic concepts and working principle of simple GA (SGA), Genetic Operators: Selection, Crossover and Mutation, flow chart of SGA, Chromosome Encoding & Decoding, Population Initialization, Objective/fitness Function, variable length Chromosome, Introduction to Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Local Search and Memetic algorithm, Application to Travelling Salesman Problem (TSP). **Evolutionary Computing-II: Multi-objective Genetic Algorithm (MOGA)):** Conflicting objectives, Objective space and variable space, Domination, Pareto front, Pareto Set, NSGA-II: Non-domination Sorting, Crowding distance operator, Applications.
Text Books, and/or reference material	**Text Books:** 1. S. Rajsekharanand and Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications", Prentice Hall of India. 2. N. P. Padhy, "Artificial Intelligence and Intelligent Systems", Oxford University Press. 3. G. Klir and B. Yuan, "Fuzzy sets and Fuzzy logic", Prentice Hall of India. 4. K. H. Lee., "First Course on Fuzzy Theory and Applications", Springer-Verlag. 5. G. J. Klir and T. A. Folger: Fuzzy Sets, Uncertainty, and Information, PH. 6. J. Yen and R. Langari, "Fuzzy Logic, Intelligence, Control and Information", Pearson Education. 7. D. Goldberg: Introduction to Genetic Algorithm. **Reference Books:** 1. Siman Haykin, "Neural Networks", Prentice Hall of India. 2. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", Wiley India. 3. Kumar Satish, "Neural Networks", Tata Mc. Graw Hill. 4. B. Yegnanarayana , "Artificial Neural Networks" 5. A. Konar, "Computational Intelligence", Springer. 6. Y. H. Pao: Adaptive Pattern Recognition and Neural Networks, Addison-Wesley.

CSO844 Compiler Design 3-0-0 3Credits 3Hours

<table>
<tr><th colspan="8">Department of Computer Science and Engineering</th></tr>
<tr><td rowspan="2">Course Code</td><td rowspan="2">Title of the course</td><td rowspan="2">Program Core (PCR) / Electives (PEL)</td><td colspan="4">Total Number of contact hours</td><td rowspan="2">Credit</td></tr>
<tr><td>Lecture (L)</td><td>Tutorial (T)</td><td>Practical (P)</td><td>Total Hours</td></tr>
<tr><td>CSO844</td><td>Compiler Design</td><td>PEL</td><td>3</td><td>0</td><td>0</td><td>3</td><td>3</td></tr>
<tr><td colspan="3">Pre-requisites</td><td colspan="5">Course Assessment methods (Continuous Assessment (CA), Mid-Term (MT), End Term (ET))</td></tr>
<tr><td colspan="3">CSC-01 (Introduction to Computing)</td><td colspan="5">CA+ MT + ET [CA: 15%, MT: 25%, ET: 60%]</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">● CO1: Able to understand the fundamental idea of compiler.
● CO2: Able to implement a part of a compiler.
● CO3: Able to know how a compiler recover from error.</td></tr>
<tr><td>Topics Covered</td><td colspan="7">● Introduction to Regular Expressions, NFA and DFA. 3L
● Introduction to the philosophy of compilers and course Overview. Introducing different phases of compilers with an example. 1L
● Details of Lexical analysis phase. Implementation of a Lexical analyzer. 4L
● Regular expression versus Grammars. Different types of Top-Down parsing. Different types of Bottom -up parsing. Implementing one Bottom -up parsing algorithm. 12L
● Introduction to Error Recovery Routine, Type Checking and Symbol Table. Introduction to lex and yacc. 4L
● Syntax Directed Translation scheme. 6L
● Intermediate code generation. Three Address Codes. 5L
● Code generation and code optimization. 5L
● Linker, Loader 2L</td></tr>
<tr><td>Text Books, and/or reference material</td><td colspan="7">Text Books: Compilers: Principles, Techniques, and Tools (Latest Edition). Alfred Aho, Monica Lam, Ravi Sethi, and Jeffrey Ullman. Addison-Wesley

Reference Books: Engineering a Compiler. Keith Cooper and Linda Torczon. Morgan Kaufman</td></tr>
</table>