DIT UNIVERSITY
Dehradun

Detailed Course Structure & Syllabus of B.Tech. – Computer Science & Engineeringwith Specializations in

AI, Machine Learning and RoboticsData Science and Analytics
Internet of Things and Edge ComputingCybersecurity and Privacy
Computer Vision and Biometrics Cloud Computing and Block chain
Full Stack and DevOps

Approved by the Academic Council in its 22nd Meeting held on 06.03.2023

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2024-28

Introduction

The Ministry of Human Resource Development (MHRD), Govt. of India, has initiated development ofa New Education Policy (NEP) to bring out comprehensive reforms in the Indian education system.

The University Grants Commission (UGC) has subsequently initiated several steps to foster academic excellence through introduction of paradigm shift in learning and teaching pedagogy, innovation and improvement in course curricula, examination and education system.

While a majority of education institutions have started following the semester-based system of education, it has been observed that this new system is still producing graduates who lack knowledge, values, skills and are not job ready professional. The reason for this lacking could be attributed to the rigidity of our program structures and lack of flexibility to have choices among core subject education, liberal arts, ability enhancement, skill development, etc., that is fundamental to overall development and employability of these graduates.

To make this possible, a fully flexible choice-based credit system (FFCBCS), a well-established internationally known system, is proposed. This fully flexible choice-based credit system allows students the flexibility to learn at their own pace, and register for both core subjects and a variety of courses from other areas, leading to holistic development of an individual. The FFCBCS will facilitate us to bench mark our programs with best international liberal arts based academic programs.

Advantages of the FFCBCS structure:

- Shift in focus from the teacher-centric to student-centric education. Student can curve out their program structure by choosing minimum number of credits from well-defined baskets.
- Student may undertake as many credits as they can cope with.
- FFCBCS allows students to choose courses from various baskets of inter-disciplinary, intra-disciplinary, skill oriented, ability enhancing, and from other disciplines.

Features unique to DIT University FFCBCS structure:

1. A minimum of 150-160 credits has to be earned by a student to be eligible for an Under Graduate degree in Engineering. Each department will decide their total credits for each program, and it can vary across disciplines.

2. Courses are categorized into 11 baskets, and a student will have the option to choose coursesin most baskets and earn ***minimum number of credits*** required in each basket for the award of his/her degree. For each basket, Engineering departments have the flexibility to identify course(s) which will be a core requirement for their program.

3. In certain disciplines, students may choose a ***Specialization*** by earning 18 credits of Discipline Elective courses towards a particular area of that discipline (intra-disciplinary). In addition to this, brighter students will have the option to receive (a) a ***Certificate*** by earning *additional* 9 credits towards a particular area either inside or outside their discipline, or (b) ***Minor*** by earning additional 18 credits towards a particular area outside their discipline. Certificates and Minors can be earned through either University courses, or with MOOCs from providers as identified by the University. Each department will design the structures and eligibility conditions for registration to its certificates or minor program, which may be reviewed annually, to keep the ***Certificates*** and ***Minors*** contemporary and relevant to latest changes.

4. An Academic Advisory Committee may be formed comprising all HoDs/ Programme Coordinator and one representative each from respective departments. Academic Advisory Committee will meet at the end of every semester after the completion of Board of Examination meeting to discuss and finalize course offerings by respective departments in the upcoming semester. Academic Advisory Committee will be chaired by the Dean Academic Affairs/ Deans of respective Schools/Competent Authority.

5. To provide sufficient flexibility and room during the program for additional ***Certificates***, ***Specializations, and Minors***, 8-week summer semesters (Summer 1, Summer 2, and Summer 3) may have to run. Summer semesters are critical for implementing a fully flexible system. Each department will decide *a priori* which courses to offer in the summer semester and get them finalized at the Academic Advisory Committee meeting.
6. Project based learning has to be incorporated as a core component of evaluation in each course, and depending on the level and type of the course, the project can be of several types - Study Oriented Project, Lab Oriented Project, Design Oriented Project, Computer Oriented Project, Projects of Organizational Aspects, Research Projects, or Entrepreneurship and Start Up Projects. A Capstone Project has been introduced in the 8th semester for all Bachelor of Technology students.
7. Courses under each basket may be updated on an annual basis.

1. Each student will be advised by a faculty advisor of his/her department for registration of courses from each basket in the beginning of semester, depending upon the availability of seats. A student advising centre may be formed where students will have access to department faculty advisers. Faculty advisers should have complete access to view individual student‘s academic transcript for advising purposes.

2. A student getting an F grade in a core course (departmental or otherwise) at the end of the semester will have to earn those credits by registering for the same course whenever it is offered in subsequent semesters. If the course is not a core course, the student may choose to register for any other course next semester in that basket as advised by the department faculty adviser. Additional fees for those number of credits may apply.

3. Students may opt for summer training/internships/industrial tours as advised by the department. However, these activities will not have credits.

Baskets of FFCBCS

11 Baskets of courses have been identified to provide student comprehensive exposure to a large number of areas, leading to the holistic development of an individual. These baskets are as follows:

1. **Language and Literature:** These include courses related to English or other popular languages worldwide, communication skills, and literature. These courses are of 3 credits each.
2. **Core Science:** These courses include science courses from the disciplines of Physics and Chemistry. These courses are of 5 credits each.
3. **Core Mathematics:** This basket includes courses from Mathematics department, crafted for Engineering students. These courses are of 4 credits each.
4. **Engineering Sciences:** This basket includes introductory courses from various disciplines of Engineering designed to provide the student solid foundation to the domain of engineering. These courses are of 4 credits each.
5. **Discipline Core:** This basket includes compulsory courses in the discipline in which the student is admitted to the University. These courses are of 4 credits each.
6. **Discipline Elective:** This basket provides students courses other than discipline core, and are normally in certain specialized areas. These courses are of 3 credits each.
7. **Humanities and Liberal Arts:** This basket includes liberal arts courses in various disciplines like psychology, management, economics, etc., and are of 3 credits each.
8. **Skill Enhancement:** Courses in this basket are primarily hands-on and aims to allow students acquire skills required in certain disciplines that are currently in high demand in the job market. These courses are of 2 credits each.

9. **Ability Enhancement:** These courses aim to enhance knowledge and ability of an individual in certain required areas related to national and societal interest. Courses in this basket are of 2 credits each.
10. **Free Electives:** Student can register for any three courses outside their department of his/her choice. These courses can also be taken from MOOCs, and a minimum of 9 credits have to be taken bya student in this basket. These courses are of 3 credits each.
11. **Capstone Project:** Capstone project is a semester long multifaceted experimental/research assignment that serves as a culminating academic and intellectual experience for students, taken in the last semester of study. It is of 12 credits and may be done groups of not more than three students, and in three modes as follows:

- **Mode A**: Project with a department faculty.
- **Mode B**: Project as part of Industry Internship arranged only by the career and placement service of the University. Students securing this assignment on their own will not be allowed, unless the project is secured at a well-known industry, and duly approved by the department. The department's decision in all such cases will be final.
- **Mode C**: Semester long project in an academic institute/lab of National/International Importance, secured by students on their own. The department's decision to allow in all such cases willbe final.

A separate rule booklet will be released for implementation of Capstone Project.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2024-28

Structure of the B.Tech. FFCBCS Program in Computer Science and Engineering & IT

Basket/Area	Min Credits To be taken	Credit per course	Courses
Language and Literature (LL) Core: Professional Communication Elective: Choose any 1 more LL course	6	3	2
Core Sciences (CoS) Core: None Elective: Choose any 2 CoS Course	10	5	2
Core Mathematics (CM) Core: Engg. Maths 1, Engg. Maths 2, Probs and Stats Elective: None	12	4	3
Engineering Sciences (ES) Core: Prog. For Problem Solving, Data Structures Elective: Choose any 3 more ES courses	20	4	5
Discipline Core (DC) Core: CO, DM, IJP, AJP, OS, DBMS, DAA, CN, SE, TOC, CD, AI Elective: None	48	4	12
Discipline Elective (DE) Core: None Elective: Choose any 6 courses as per your Specialization	18	3	6
Humanities and Liberal Arts (HL) Core: None Elective: Choose any 3 HL Courses	9	3	3
Skill Enhancement Courses (SEC)* Core: None Elective: Choose any courses to complete credits	8	-	-
Ability Enhancement Courses (AEC)* Core: Entrepreneurship and startups, Env. Sc, Indian Constitution Elective: None	8	-	-
Free Electives (FE) Core: None Elective: Choose any 3 FE courses	9	3	3
Capstone Project (TP) Mode A: Project with a department faculty Mode B: Project as part of Industry Internship Mode C: Project in an academic institute/lab of National Importance. All Modes must be semester long	12	12	1
Total Credits	160		

* *Credits in SEC and AEC courses may vary.*

Approved by the Academic Council in its 22nd Meeting held on 06.03.2023

DIT University CSE/IT FFCBCS Program Structure

Basket/Area	Credits
Language and Literature (LL)	6
Core Sciences (CoS)	10
Core Mathematics (CM)	12
Engineering Sciences (ES)	20
Discipline Core (DC)	48
Discipline Elective (DE)	18
Humanities and Social Sciences (HSS)	9
Skill Enhancement Courses (SEC)	8
Ability Enhancement Courses (AEC)	8
Free Electives (FE)	9
Project (PRJ)	12
Total	**160**

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2024-28

Course Baskets: University FFCBCS Baskets (other than DC/DE) for B.Tech Programs
A * against a course means it is a core course for all B.Tech. Students.

Course Code	FFCBCS Baskets (other than DC/DE)				
	Language and Literature (min 6 credits to be taken)	**Contact Hrs**			**Credits**
	Name of Courses	L	T	P	C
LAF181	Professional Communication*	2	0	2	3
LAF182	Indian English Literature	3	0	0	3
LAF183	English Language Teaching	3	0	0	3
LAF184	Corporate Communication and Soft Skills	2	0	2	3
	Core Sciences (min 10 credits to be taken)				
	Name of Courses	L	T	P	C
CHF101	Engineering Chemistry(CSE,IT,EE,ECE)	3	1	2	5
CHF102	Applied Engineering Chemistry (for ME/CE/PE)	3	1	2	5
PYF101	Wave & Optics and Introduction to Quantum Mechanics	3	1	2	5
PYF102	Introduction to Mechanics	3	1	2	5
PYF103	Electricity & Magnetism	3	1	2	5
PYF105	Engineering Physics* (Since 2022)	3	1	2	5
	Core Mathematics (min 8 credits to be taken)				
	Name of Courses	L	T	P	C
MAF101	Engineering Mathematics I*	3	1	0	4
MAF102	Engineering Mathematics II*	3	1	0	4
MAF201	Engineering Mathematics III (EE, ME, CE)	3	1	0	4
MAF202	Probability and Statistics (CSE, IT, ECE, PE)	3	1	0	4
	Engineering Sciences (min 20 credits to be taken)				
	Name of Courses	L	T	P	C
ECF101	Fundamental of Electronics Engineering.	3	0	2	4
EEF101	Basic Electrical Engineering	3	0	2	4
EEF143	Electrical and Electronics Engineering Practice (non EE/EECE)	3	0	2	4
MEF101	Thermodynamics	3	1	0	4
CSF101	Programming for Problem Solving*	3	0	2	4
CSF102	Data Structures*	3	0	2	4
MEF102	Engineering Graphics	2	0	4	4
MEF103	Engineering Mechanics	2	1	2	4
MEF106	Modern Manufacturing techniques	2	0	4	4
MEF201	Mechanical Engineering Materials	3	0	2	4
PEF204	Fluid Mechanics	3	0	2	4
EEF141	Electrical Engineering Material	3	1	0	4
ECF142	Fundamental of Semiconductor Electronics	3	1	0	4
ECF144	Digital Electronics and Applications	3	0	2	4
CEF101	Civil Engineering Materials	3	1	0	4

Approved by the Academic Council in its 22nd Meeting held on 06.03.2023

	Skill Enhancement (min 8 credits to be taken)				
	Name of Courses	L	T	P	C
CSF 306	Technical Training 1	2	0	4	4
CSF 307	Technical Training 2	2	0	4	4
CSFXXX	Value Added Training 1	0	0	4	2
CSFXXX	Value Added Training 2	0	0	4	2
SWAYXXX	MOOCs (as advised by the departments)	2	0	0	0
	Ability Enhancement (min 8 credits to be taken)				
	Name of Courses	L	T	P	C
CHF201	Environmental Science*	2	0	0	2
LAF285	Indian Constitution*	2	0	0	2
MEF483	Entrepreneurship and Start-ups*	0	0	4	2
UCF201	Aptitude and Soft Skills	2	0	0	2
	Humanities and Liberal Arts (min 9 credits to be taken)				
	Name of Courses	L	T	P	C
LAF281	Introduction to Psychology	3	0	0	3
LAF282	Human Values	3	0	0	3
LAF283	Literature, Language & Society	3	0	0	3
LAF284	Principles of Management	3	0	0	3
LAF285	Indian Constitution	2	0	0	2
LAF286	Youth Psychology	3	0	0	3
LAF287	Sustainable Development	3	0	0	3
LAF381	Positive Psychology & Living	3	0	0	3
LAF382	Engineering Economics	3	0	0	3
LAF383	Introduction to Linguistics	2	0	2	3
LAF384	Creative Writing	3	0	0	3
LAF385	Health Psychology	3	0	0	3
LAF386	Ecology and Human Development	3	0	0	3
LAF481	Application of Psychology	3	0	0	3
LAF482	Intellectual Property Rights	3	0	0	3
LAF483	Science Technology and Society	3	0	0	3
LAF484	Education and Social Change	3	0	0	3
LAF485	Industrial Psychology	3	0	0	3
LAF486	Innovation and Entrepreneurship	3	0	0	3
	Free Electives (min 9 credits to be taken)				
	Name of Courses	L	T	P	C
ECF481	Analog Electronics (ECE)	3	0	0	3
ECF482	Cellular Communication Network (ECE)	3	0	0	3
ECF381	Microcontroller (ECE)	3	0	0	3
ECF382	Bio Medical Instrumentation (ECE)	3	0	0	3
ECF483	Digital Image processing (ECE)	3	0	0	3
CSF381	Software Project Management	3	0	0	3
CSF345	Introduction to Data Science	3	0	0	3

CSF482	Introduction to Cybersecurity	3	0	0	3
MEF348	Robotics Engineering	3	0	0	3
MEF381	Composites materials	3	0	0	3
MFF444	Operation Research	2	1	0	3
MEF446	Product Design & Development	3	0	0	3
MEF481	Total Quality Management	3	0	0	3
MEF482	Renewable Energy Sources	3	0	0	3
MEF485	Solar Energy System	2	0	2	3
PEF 381	Carbon Capture and Sequestration	3	0	0	3
PEF 491	Polymer Technology	3	0	0	3
PEF 492	Health, Safety and Environment in Industry	3	0	0	3
CEF281	Properties of Materials	3	0	0	3
CEF382	Disaster Preparedness Planning & Management	3	0	0	3
CEF481	Environmental Management & Sustainability	3	0	0	3
CEF482	Natural Dynamics	3	0	0	3
CEF483	GIS	3	0	0	3
CEF484	Resource Dynamics and Economic Implications	3	0	0	3
CEF343	Environmental Risk Assessment and Disaster Management	3	0	0	3
CEF348	Air & Water Pollution	3	0	0	3
CEF349	Remote sensing and Image processing	3	0	0	3
CHF366	Green Chemistry	3	0	0	3
MAF452	Optimization Techniques	3	0	0	3
	Project (12 credits)				
UCF439	Capstone Project	0	0	12	12

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2024-28

Course Baskets: B.Tech. CSE and IT FFCBCS DC Basket and CSE DE specialized tracks Baskets. IT DE courses will be chosen from these baskets.

Discipline Core (48 credits)						
	Contact Hrs Credits					
	Name of Courses	**Pre-requisite Courses**	L	T	P	C
CSF201	Computer Organization and Architecture	None	3	1	0	4
CSF202	Discrete Mathematics	None	3	1	0	4
CSF203	Introduction to Java Programming	CSF101	3	0	2	4
CSF204	Operating Systems	CSF201	3	0	2	4
CSF205	Database Management Systems	CSF101	3	0	2	4
CSF206	Advanced Java Programming	CSF203	3	0	2	4
CSF301	Software Engineering	CSF102	3	0	2	4
CSF302	Design and Analysis of Algorithms	CSF102	3	0	2	4
CSF303	Computer Networks	CSF201	3	0	2	4
CSF304	Artificial Intelligence	CSF102	3	0	2	4
CSF305	Theory of Computation	CSF202	3	1	0	4
CSF401	Compiler Design	CSF202	3	0	2	4
Discipline Electives (min 18 credits to be taken)						
Artificial Intelligence, Machine Learning and Robotics						
	Name of Courses		L	T	P	C
CSF341	R Programming		2	0	2	3
CSF342	Fuzzy Logic and Neural Network		2	0	2	3
CSF343	Evolutionary Computing		2	0	2	3
CSF344	Machine Learning		2	0	2	3
CSF441	Deep Learning		2	0	2	3
CSF442	Robotics		2	0	2	3
Data Science and Analytics						
CSF341	R Programming		2	0	2	3
CSF344	Machine Learning		2	0	2	3
CSF345	Introduction to Data Science		2	0	2	3
CSF346	Data Mining and Data Warehousing		2	0	2	3
CSF441	Deep Learning		2	0	2	3
CSF443	Big Data Analytics		2	0	2	3
Internet of Things and Edge Computing						
CSF347	Wireless and Mobile Systems		3	0	0	3
CSF348	Mobile Application Programming using Android		2	0	2	3
CSF349	Cloud Computing		2	0	2	3
CSF351	Advanced Computer Networks		3	0	0	3
CSF444	Internet of Things		2	0	2	3
CSF445	Mobile & Wireless Network Security		2	0	2	3
Cyber security and Privacy						
CSF352	Number Theory and Cryptology		2	0	2	3

Approved by the Academic Council in its 22nd Meeting held on 06.03.2023

CSF353	Foundation of Cyber Security	2	1	0	3
CSF354	Data Encryption & Network Security	2	0	2	3
CSF355	Cyber Crime & Investigation	2	0	2	3
CSF445	Mobile & Wireless Network Security	2	0	2	3
CSF446	Ethical Hacking & Digital Forensics	2	0	2	3
Computer Vision and Biometrics					
CSF341	Machine Learning	2	0	2	3
CSF356	Digital Image Processing	2	0	2	3
CSF357	Satellite Image Processing	2	0	2	3
CSF358	Computer Vision	2	0	2	3
CSF447	Information Retrieval	2	0	2	3
CSF448	Biometrics Security	3	0	0	3
Cloud Computing and Blockchain					
CSF349	Cloud Computing	2	0	2	3
CSF354	Data Encryption and Network Security	2	0	2	3
CSF361	Introduction to Blockchain Technologies	2	1	0	3
CSF362	Design & Development of Blockchain Technologies	2	0	2	3
CSF363	Blockchain Ecosystems & Governance	2	1	0	3
CSF364	Container Technologies	2	0	2	3
Full Stack and DevOps					
CSF349	Cloud Computing	2	0	2	3
CSF364	Container Technologies	2	0	2	3
CSF371	Front-End Engineering	2	0	2	3
CSF372	Advance Topics in Front-End Engineering	2	0	2	3
CSF373	Server Side Engineering	2	0	2	3
CSF374	DevOps	2	0	2	3

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2024-28

Abbreviations

1	PSP	Problem Solving and Programming
2	DS	Data Structures
3	COA	Computer Organization & Architecture
4	OS	Operating System
5	DAA	Design and Analysis of Algorithm
6	AI	Artificial Intelligence
7	DM	Discrete Mathematics
8	DBMS	Data Base Management Systems
9	SE	Software Engineering
10	TOC	Theory of Computation
11	CD	Compiler Design
12	IJP	Introduction to Java Programming
13	AJP	Advanced Java Programming
14	CN	Computer Networks

Flow of Actions for implementing FFCBCS every semester

After release of Final Exam results, Academic Advisory Committee meets to decide & finalize course offerings in each basket

↓

Courses are created in ERP with required number of seats

↓

Registrar announces the date for Registration

↓

Students get advised and registers for courses in the Student Advising Centre

↓

List of students gets added in LMS

↓

Class Starts

Approved by the Academic Council in its 22nd Meeting held on 06.03.2023

1.	School offering the course	School of Computing
2.	Course Code	**CSF201**
3.	Course Title	**Computer Organization & Architecture**
4.	Credits (L: T:P:C)	3:1:0:4
5.	Contact Hours (L: T:P)	3:1:0
6.	Prerequisites (if any)	CSF101
7.	Course Basket	Discipline Core

COURSE SUMMARY

The course is proposed to teach the students the concepts of computer organization for several engineering computing systems. Students will develop the ability and confidence to use the fundamentals of computer organization as a tool in the engineering of digital systems.

COURSE OBJECTIVES

This course will facilitate the students to learn the fundamentals of computer organization and its relevance to classical and modern problems of computer design.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: This will help the students to be familiarized with the hardware components and concepts related to the control design, data representation and evaluation process of different arithmetic operations.

CO2: This will help the students to be familiarized with CPU organization addressing modes, different types of instruction formats.

CO3: The student will be able to learn the hardware components and concepts related to the input/output and memory organization.

CO4: Students will be able to get the theoretical concept of parallel processing and different types of multiprocessor's interconnection structures.

CURRICULUM CONTENT

Unit 1: Basic Structure of Computers & Register Transfer Language (8 L)

Basic Structure of Computers: Computer Types; Functional Units

Register Transfer and Micro operation: Register Transfer Language, Bus and Memory Transfers, Bus Architecture, Arithmetic, Logic, Shift Micro-operation, Design of ALU.

Unit 2: Computer Arithmetic (8 L)

Introduction, Addition and Subtraction Algorithms, Multiplication and Division Algorithms, Floating Point Arithmetic Operation, IEEE Format for Floating Point Numbers.

Unit 3 Control Design & Processor Organization (8 L)

Control Design: Execution of a Complete Instruction, Sequencing of Control Signals, Single and Multiple Bus Architecture, Hardwired Control Unit, Micro Programmed Control Unit

Processor Organization:

Accumulator Organization General Register Organization, Stack Organization, Addressing Modes, Instruction Format, Data Transfer & Manipulations, Program Control.

Unit-4 Input-Output & Memory Organization (8 L)

Input-Output Organization: I/O Interface, Modes of Transfer, Interrupts & Interrupt Handling, Direct Memory Access, Input-Output Processor, Serial Communication.

Memory Organization: Memory Hierarchy, Main Memory (RAM And ROM Chips), Organization of Cache Memory (performance and mapping), Virtual Memory, Page Replacement Techniques.

Unit- 5: Parallel Processing & Multiprocessor (8 L)

CPU Performance: Processor Clock, Clock Rate, Cycle, Basic Performance Equation, and MIPS Rate.

Parallel Processing: Flynn's classification, Pipelining- Arithmetic Pipelining, Vector Processing, Array Processor, pipeline hazards.

Multiprocessor: Characteristic of Multiprocessor, Interconnection Structure, Cache Coherence.

TEXT BOOKS:

1. John P. Hayes. Computer Architecture and Organization, 4th Edition, McGraw Hill, 2010.
2. M. Morris Mano. Computer System Architecture 3rd Ed, Pearson.
3. Carl Hamacher, ZvonkoVranesic, Safwatzaky. Computer Organization, 5th Edition.

REFERENCES :

1. John L. Hennessey and David A. Patterson: Computer Architecture, A Quantitative Approach, 4th Edition, Elsevier, 2007.
2. Kai Hwang: Advanced Computer Architecture Parallelism, Scalability, Programmability, 2nd Edition, Tata Mc Graw Hill, 2010.

1.	School offering the course	School of Computing
2.	Course Code	**CSF202**
3.	Course Title	**Discrete Mathematics**
4.	Credits (L: T:P:C)	3:1:0:4
5.	Contact Hours (L: T:P)	3:1:0
6.	Prerequisites (if any)	None
7.	Course Basket	Discipline Core

COURSE SUMMARY

This course covers elementary discrete mathematics for computer science and engineering. It emphasizes mathematical definitions and proofs as well as applicable methods. Topics include formal logic notation, proof methods; induction, well-ordering; sets, relations; elementary graph theory; integer congruence; asymptotic notation and growth of functions; permutations and combinations, and counting principles.

COURSE OBJECTIVES

The objectives of this course is to learn concepts of Discrete Mathematics and by applying the algorithms to solve the problems related to Recursion, combinatorial mathematics and problems on basic graph theory

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Apply the techniques to perform the operations on discrete structures such as sets, functions, relations, and sequences.
CO2: Identify the properties of Lattice by constructing the Hasse Diagram and demonstrate the proofs to solve problems using counting techniques.
CO3: Apply the properties of Algebric structures and design the propositional and predicate logic.
CO4: Apply the properties of Graph and Recurrence Relation to solve computational problems.

CURRICULUM CONTENT

UNIT I: Introduction to Sets, Relations & Functions (7 L)

Set Theory: Introduction, Combination of sets, Multisets, ordered pairs, Set Identities.

Relations: Definition, Operations on relations, Properties of relations, Composite Relations, Equality of relations, Order of relations.

Functions: Definition, Classification of functions, Operations on functions, Recursively defined functions. Natural Numbers: Introduction, Mathematical Induction

UNIT II: Posets & Introduction to Boolean algebra (6 L)

Partial order sets: Definition, Partial order sets, Combination of partial order sets, Hasse diagram. Lattices: Definition, Properties of lattices Bounded, Complemented and Complete Lattice Combinatorics: Introduction, Counting Techniques, Pigeonhole Principle.

UNIT III: Groups & Rings (8 L)

Algebraic Structures: Definition, Groups, Subgroups and order, Cyclic Groups, Cosets, Lagrange's theorem, Normal Subgroups, Definition and elementary properties of Rings and Fields, Integers modulo n.

UNIT IV: Propositional logic, Predicate Logic & Introduction to Probability (8 L)

Propositional Logic: Proposition, well-formed formula, Truth tables, Tautology, Contradiction, Algebra of proposition, Theory of Inference, Natural Deduction.

Predicate Logic: First order predicate, well-formed formula of predicate, quantifiers, Inference theory of predicate logic.

UNIT V: Introduction to Graphs & Recurrence Relations (8 L)

Graphs: Definition and terminology, Representation of graphs, multigraphs, bipartite graphs, Planar graphs, Isomorphism and Homeomorphism of graphs, Euler and Hamiltonian paths, Graph coloring. Recurrence Relation & Generating function: Recursive definition of functions, Recursive algorithms, Method of solving recurrences.

TEXT BOOKS:

1. Liu C.L., Elements of Discrete Mathematics, McGraw Hill Education. 4th edition2017.
2. Kolman B & Busby C.R., Discrete Mathematical Structure for Computer Science, Prentice Hall of India Ltd. 6th Edition 2008.
3. Deo N., Graph Theory, Prentice Hall of India 1974

REFERENCE BOOKS :

1. Trembley J.P. &Manohar R., Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill.1st Indian Edition 2017

1.	School offering the course	School of Computing
2.	Course Code	**CSF203**
3.	Course Title	**Introduction to Java Programming**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF101
7.	Course Basket	Discipline Core

COURSE SUMMARY

This course covers Java and fundamental programming techniques with primitive data types, variables, constants, assignments, expressions, and operators, selection statements, mathematical functions, characters, and strings, loops, methods, and arrays. Students will learn how to write recursive methods for solving inherently recursive problems. The next part will introduce object-oriented programming. Java is an object-oriented programming language that uses abstraction, encapsulation, inheritance, and polymorphism to provide great flexibility, modularity, and reusability in developing software. Students will learn programming with objects and classes, class inheritance, polymorphism, exception handling, abstract classes, interfaces, Text I/O and binary I/O.

COURSE OBJECTIVES

The objectives of this course are to learn object oriented programming paradigm using Java as programming language. Students will be exposed to fundamental concepts in java programming language, followed by object oriented paradigm and its building blocks.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Understand and implement fundamental programming techniques and data types, variables, constants, assignments, expressions, and operators of Java programming language.

CO2: Understand and implement selection statements, mathematical functions, characters, strings, loops.

CO3: Understand and implement methods, arrays and recursion using Java.

CO4: Understand and implement object-oriented paradigm using objects and classes, abstraction, encapsulation, inheritance, polymorphism, interfaces, and exception handling.

CURRICULUM CONTENT

UNIT I: Introduction, Fundamental Programming Techniques (6 L)

Introduction, the Java Language Specification, API, JDK, and IDE, Creating, Compiling, and Executing a Java Program, Developing Java Programs Using Net Beans. Identifiers, Variables, Assignment Statements and Assignment Expressions, Named Constants, Naming Conventions, Numeric Data Types and Operations, Numeric Literals, Evaluating Expressions and Operator Precedence, Increment and Decrement Operators, Numeric Type Conversions.

UNIT 2: Selection Statements, Loops, Mathematical Functions, Characters and Strings (6 L)

Boolean Data Type, if Statements, Two-Way if-else Statements, Nested if and Multi-Way if-else Statements, Logical Operators, switch Statements, Conditional Expressions, Operator Precedence and Associativity. Common Mathematical Functions, Character Data Type and Operations, the String Type.

The while Loop, the do-while Loop, The for Loop, Nested Loops, Keywords break and continue.

UNIT 3: Methods, Arrays and Recursions (6 L)

Defining a Method, calling a Method, Passing Arguments by Values, Modularizing Code, Overloading Methods, The Scope of Variables, Method Abstraction and Stepwise Refinement.
Array Basics, Copying Arrays, Passing Arrays to Methods, Returning an Array from a Method, Searching Arrays, Sorting Arrays, The Arrays Class.
Two-Dimensional Array Basics, Passing Two-Dimensional Arrays to Methods, Multidimensional Arrays.
Recursion, writing recursive codes in Java.

UNIT4: Object Oriented Paradigm (18 L)

Defining Classes for Objects, Constructing Objects Using Constructors, Accessing Objects via Reference Variables, Using Classes from the Java Library, Static Variables, Constants, and Methods, Visibility Modifiers, Data Field Encapsulation, Passing Objects to Methods, Array of Objects, Immutable Objects and Classes, This Reference.

Class Abstraction and Encapsulation, Thinking in Objects, Processing Primitive Data Type Values as Objects, Types and, The Big Integer and Big Decimal Classes, The String Class, The String Builder and String Buffer Classes.
Superclass and Subclasses, Using the super Keyword, Overriding Methods, Overriding vs. Overloading, The Object Class and Its to String( ) Method, Polymorphism, Dynamic Binding, Casting Objects and the instance of Operator, The Object's equals Method.

Exception-Handling Overview, Exception Types, the finally Clause, When to Use Exceptions, Defining Custom Exception Classes.

Abstract Classes, Interfaces, The Comparable Interface, The Clone able Interface, Interfaces vs. Abstract Classes.

TEXTBOOK(S)

1. Intro to Java Programming (Comprehensive Version), by Y. Daniel Liang. Publisher: Pearson Education; Tenth edition (2018), ISBN-10: 935306578X, ISBN-13: 978-9353065782

REFERENCE BOOKS

1. Java - The Complete Reference, by Herbert Schildt, Publisher: McGraw Hill Education; Tenth edition (2017), ISBN-10: 9789387432291, ISBN-13: 978-9387432291

List of Experiments

S.NO.	EXPERIMENT NAME
1	Program in Java to design simple calculator for (+, -, *, and /) using switch case
2	Program in Java to design accounts class and two functions withdraw () and deposit ().
3	Program in Java to search a particular element in a one dimensional array.
4	Program in Java to the concept of polymorphism by designing functions to sum different type of numbers
5	Program to show the concept of method overriding in Java.
6	Program in Java that import the user define package and access the Member variable of classes that Contained by Package.
8	Program in Java to handle the Exception using try and multiple catch block.
9	Program in Java demonstrating usage of abstract classes and interfaces.
10	Write a program to demonstrate usage of constructor chaining in inheritance.

1.	School offering the course	School of Computing
2.	Course Code	**CSF204**
3.	Course Title	**Operating System**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF201
7.	Course Basket	Discipline Core

COURSE SUMMARY

This course will introduce the core concepts of operating systems, such as processes and threads, scheduling, synchronization, memory management, file systems, input and output device management and security.

COURSE OBJECTIVES

This course is classified into two sections: a theory section that educates to students about the theories and principles that underlie modern operating systems, and a practical section that relates theoretical principles to operating system implementation. Theory section basically includes: Process and processor management, concurrency and synchronisation, memory management schemes, file system and secondary storage management, etc.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Describe the basic concepts of operating systems, including development and achievements, functionalities and objectives, structure and components.

CO2: Understand the general architecture & functioning of operating system such as processes, threads, files, Concurrency, IPC abstractions, shared memory regions, etc.

CO3: Analyze various algorithms eg. Process scheduling and memory management algorithms.

CO4: Categorize the operating system's resource management techniques, deadlock management techniques, memory management techniques.

CURRICULUM CONTENT

Unit 1: Introduction to Operating System (8 L)

Introduction: Components of a computer System, Operating system: User view & System view, Evolution of operating system, Single Processor & Multiprocessor systems, Real Time System, Distributed Systems, Multimedia Systems, Handheld Systems.

Operating System Structure: Operating System Services, User Operating System Interfaces: Command- Line and GUI, System Calls.

Unit 2: Management & Scheduling (6 L)

Process Management: Process Concept, Process States, Process Transition Diagram, Process Control Block (PCB). CPU Scheduling: Scheduling Concepts, Performance Criteria, Scheduling Queues, Schedulers, Scheduling Algorithms: Preemptive &Non Preemptive: FCFS, SJF, Priority, Round-Robin.

Unit 3: Concurrent Processes & Deadlocks (8 L)

Concurrent Processes: Principle of Concurrency, Producer / Consumer Problem, Co-operating Processes, Race Condition, Critical Section Problem, Peterson's solution, Semaphores, Classical Problem in Concurrency- Dining Philosopher Problem; Inter Process Communication models and Schemes.

Deadlock: System Model, Deadlock Characterization, Prevention, Avoidance and Detection, Recovery from deadlock.

Unit 4: Memory Management **(7 L)**

Memory Management: Bare machine, Resident monitor, Multiprogramming with fixed partition, Multiprogramming with variable partition, Multiple base register, Paging, Segmentation, Virtual memory concept, Demand paging, Performance, Paged replaced algorithm, Allocation of frames, Cache memory.

Unit- 5: File System & I/O Management **(7 L)**

File System: Different types of files and their access methods, various allocation methods. I/O Management and Disk Scheduling: I/O Devices, Organization of I/O functions, Disk Structure, Disk Scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK).

TEXT BOOKS:

1. Silberschatz, Galvin and Gagne, ―Operating Systems Concepts‖, Wiley, 9th Edition 2018.

REFERENCES:

1. Harvey M. Dietel, ― An Introduction to Operating System‖, Pearson Education 1st Edition 2009.
2. D M Dhamdhere, ―Operating Systems: A Concept based Approach‖, PHI. 3rd Edition.2017.

List of Experiments

S.NO.	EXPERIMENT NAME
1	Implement the following algorithm FCFS, SJF, Round Robin, Priority in Linux.
2	Implement the concept of fork () system call using C programming in Linux environment only.
3	Implement the concept of threading in OS. Prefer threading in JAVA only.
4	Write a Java program to simulate producer-consumer problem using semaphores.
5	Write a Java program to simulate the concept of Dining Philosopher‘s problem.
6	Write a program using Linux to simulate Banker‘s algorithm.
7	Write a C program using Linux to simulate the following contiguous memory allocation techniques: a> Worst fit b> Best fit c> First fit.
8	Write a Java program to simulate the disk scheduling algorithms: a> FCFS b> SCAN c> C-scan
9	Write a C program using Linux to implement page replacement algorithms: a> FIFO b> LRU c> LFU
10	Write a C program to compare the Optimal page replacement algorithm with FIFO and LRU page replacement algorithms.

1.	School offering the course	School of Computing
2.	Course Code	**CSF205**
3.	Course Title	**Database Management System**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF101
7.	Course Basket	Discipline Core

COURSE SUMMARY

The students will learn the basic theory of databases. They will be able to design and develop a database using conceptual schema, logical schema, and physical schema and are expected to learn how to write database management system software. They will also learn some of the specialized databases.

COURSE OBJECTIVES

This course aims to educate students on the role of a well-structured relational database management system (RDBMS) in the efficient functioning of an organization. This course covers theory and practice in designing a relational database management system with an example of a current database product of MYSQL. Students also learn about the important concepts of database integrity, security, and availability with techniques like normalization, concurrency control, and recoverability control.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Demonstrate the basic elements of a relational database management system.
CO2: Identify the data models for relevant problems.
CO3: Design entity-relationship and convert entity-relationship diagrams into RDBMS and formulate SQL queries.
CO4: Apply and create relational database design process with Normalization and De-normalization of data so that data redundancy, data inconsistency, and data loss problems may be resolved.

CURRICULUM CONTENT

Unit 1: Introduction to Database System (7 L)

Introduction: Database System Applications, database System VS file System, Data Abstraction,
Instances and Schemas, data Models: the ER Model, Relational Model & Other Models, Database Languages, database Users and Administrator, database System Structure, Storage Manager, the Query Processor, Two/Three-tier architecture.

Unit 2: E-R modeling Data Base Design (8 L)

E-R model: Basic concepts, Design Issues, Mapping Constraints, Attributes and Entity sets, Relationships and Relationship sets, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended ER features.

Unit 3 Relational Model & SQL (7 L)

Relational Model: Structure of relational Databases, Relational Algebra, Relational Calculus, Extended Relational Algebra
SQL: Form of Basic SQL Query, Nested Queries, Aggregative Operators, NULL values, Logical operators, Outer Joins, Complex Integrity Constraints in SQL.

Unit-4 Database Design Concepts (7 L)

Database Design: Schema refinement, Different anomalies in designing a Database, Decompositions, Problem-related to decomposition, Functional Dependency, Normalization using functional dependencies, 1NF, 2NF, 3NF & BCNF, Lossless join decomposition, Dependency preserving Decomposition, Schema Refinement in Database Design, Multivalued Dependencies Closer properties of Multivalued dependency, Join dependency, 4NF, 5NF.

Unit- 5: Transaction & Concurrency **(8 L)**

Transaction Management: Transaction-concepts, states, ACID property, schedule, serializability of schedules, concurrency control techniques - locking, timestamp, deadlock handling, recovery-log based recovery, shadow paging.

TEXT BOOKS :

1. Raghurama Krishnan, Johannes Gehrke, Database Management Systems, TATA McGraw-Hill 3rd Edition,2014
2. Silberschatz, Korth, Database System Concepts, McGraw hill, 6th edition,2013
3. Elmasri Navate, Fundamentals of Database Systems, Pearson Education,7th edition 2016

REFERENCES :

1. Peter Rob & Carlos Coronel, Database Systems design, Implementation, and Management, Course Technology Inc, 7thEdition, 2006.
2. C.J. Date , Introduction to Database Systems, Pearson Education,8th edition,2012
3. Bayross I., SQL, PL/SQL the Programming Language of Oracle, BPB Publications (2009) 4th ed.
4. HofferJ., Venkataraman, R. and Topi, H., Modern Database Management, Pearson (2016) 12th edition.

List of Experiments

S.NO.	EXPERIMENT NAME
1	Implementation of Data Definition language in Query Language.
2	Implementation of Data Manipulation in Query Language.
3	Insertion & Updation of records in the database table
4	Implementation of GROUP functions (avg, count, max, min, Sum).
5	Execution of the various type of SET OPERATORS (Union, Intersect, Minus).
6	Apply the various types of Integrity Constraints on the table.
7	Creation of various types of JOINS.
8	Implementation of Views and Indices in database.
9	Implementation of the foreign key on the database.
10	Modify the database structure and drop the record with structure.

1.	School offering the course	School of Computing
2.	Course Code	**CSF206**
3.	Course Title	**Advanced Java Programming**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF203
7.	Course Basket	Discipline Core

COURSE SUMMARY

This course covers advanced Java programming concepts that includes Java user interface programming and design, collections framework, multithreading, and network programming using Java. Students will also be introduced to other editions of Java and their technologies.

COURSE OBJECTIVES

The objectives of this course are to learn advanced java programming techniques and technologies required to build applications with good user interfaces.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand and implement swing components and event handling mechanisms.
CO2: Understand and implement various collections classes and interfaces.
CO3: Understand and implement multithreading concepts using Java.
CO4: Understand and implement network programming in Java.

CURRICULUM CONTENT

UNIT I: User Interface (12 L)

Review of OOP concepts in Java, Java Swing Framework, Swing and AWT, Basics of GUI, Various Swing Components and demonstration of their usage in java programs.
Event Handling, Linking swing components to appropriate events, Listener interfaces
Graphics class, linking events to graphics objects.

UNIT 2: Java Collections Framework (6 L)

Collections, List, Array List and Linked List, Set, Hashset and Treeset, Map, Hashmap and Treemap, Iterators, the Comparator Interface, Collections class.

UNIT 3: Multithreading (9 L)

Thread Concepts, Creating Threads, The Thread Class, Runnable interface, Thread Pools, Thread Synchronization, Synchronization Using Locks, Cooperation among Threads, Semaphores, Avoiding Deadlocks using Java, Thread States, Synchronized Collections.

UNIT4: Network Programming (9 L)

Client/Server Computing, Socket, Server and Client Sockets, Data Transmission through Sockets, Input Stream, OutputStream, InetAddress Class, Serving Multiple Clients and Multithreaded Servers, Sending and Receiving Objects.
Introduction to Java Enterprise Edition and its related technologies.
Introduction to Java Micro Edition and Java ME Embedded Technologies, its usage in Internet of Things.

TEXTBOOK(S)

1. Intro to Java Programming (Comprehensive Version), by Y. Daniel Liang. Publisher: Pearson Education; Tenth edition (2018), ISBN-10: 935306578X, ISBN-13: 978-9353065782

REFERENCE BOOKS

1. Java - The Complete Reference, by Herbert Schildt, Publisher: McGraw Hill Education; Tenth edition (2017), ISBN-10: 9789387432291, ISBN-13: 978-9387432291

List of Experiments

S.NO.	EXPERIMENT NAME
1	Program in Java to implement OOP concepts like inheritance and polymorphism.
2	Program in Java to implement Swing components and link events to these components.
3	Program in Java to design simple calculator for arithmetic operations using GUI.
4	Program in Java to draw shapes using Graphics class.
5	Program in Java using implementations of Hashsets and Treesets.
6	Program in Java using implementations of Arraylist and LinkedList.
7	Program in Java using implementations of Maps.
8	Program in Java demonstrating usage of Collections.sort.
9	Program in Java to implement a Server and Client Socket using Input and Output Streams.
10	Program in Java to implement a server serving multiple clients.

1.	School offering the course	School of Computing
2.	Course Code	**CSF301**
3.	Course Title	**Software Engineering**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF102
7.	Course Basket	Discipline Core

COURSE SUMMARY

Software Engineering (SE) comprises the core principles consistent in software construction and maintenance: fundamental software processes and life-cycles, mathematical foundations of software engineering, requirements analysis, software engineering methodologies and standard notations, principles of software architecture and re-use, software quality frameworks and validation, software development, and maintenance environments and tools. An introduction to object-oriented software development process and design.

COURSE OBJECTIVES

Knowledge of basic SW engineering methods and practices, and their appropriate application. Describe software engineering layered technology and Process framework. A general understanding of software process models such as the waterfall and evolutionary models. Understanding of software requirements and the SRS documents. Understanding of the role of project management including planning, scheduling, risk management, etc. Describe data models, object models, context models and behavioral models. Understanding of different software architectural styles. Understanding of implementation issues such as modularity and coding standards. Understanding of approaches to verification and validation including static analysis, and reviews. Understanding of software testing approaches such as unit testing and integration testing. Describe software measurement and software risks. Understanding of software evolution and related issues such as version management. Understanding on quality control and how to ensure good quality software.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Apply the various design models of software engineering, and Implementation of Software Life Cycle Model.

CO2: Develop proper SRS for software quality assurance.

CO3: Demonstrate the complexities of software projects at the beginning of design phases.

CO4: Estimate the cost and budget of projects, and Removing the errors and bugs so that re-design of models can be done.

CURRICULUM CONTENT

UNIT 1 **(7 L)**

Introduction to Software Engineering: Software Characteristics, Software Crisis, Software Engineering Processes, Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.

UNIT 2 **(8 L)**

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT 3 **(8 L)**

Design Engineering: Design process and Design quality, Design concepts, the design model, pattern based software design.

Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, assessing alternative architectural designs, mapping data flow into software architecture.
Modeling component-level design: Designing class-based components, conducting component-level design, object constraint language, designing conventional components.

UNIT 4 **(9 L)**
Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.
Product metrics: Software Quality, Frame work for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Software Measurement, Metrics for software quality.

UNIT 5 **(8 L)**
Risk management: Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM.
Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software Quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXTBOOKS

1. R. S. Pressman, ―Software Engineering – A practitioner ‘s approach‖, McGraw Hill Education; 7 Edition (2009)
2. K.K. Aggarwal & Yogesh Singh, ―Software Engineering‖, New Age International, 2nd Ed. 2006.
3. Pankaj Jalote, Software Engineering, Wiley India, 2010

REFERENCE BOOKS

1. Rajib Mall, Fundamentals of Software Engineering, PHI Publication, 4th Edition, 2014.
2. Ian Sommerville, Software Engineering, Addison Wesley, 10th Edition, 2015
3. James Peter, W Pedrycz, ―Software Engineering‖, John Wiley & Sons,2000

List of Experiments

S.NO.	EXPERIMENT NAME
1	Development of problem statement.
2	Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3	Preparation of Software Configuration Management and Risk Management related documents.
4	Study and usage of any Design phase CASE tool
5	Performing the Design by using any Design phase CASE tools.
6	Develop test cases for unit testing and integration testing
7	Develop test cases for various white box and black box testing techniques.
	Sample Projects:
8	1. Passport automation System 2. Book Bank 3. Online Exam Registration
9	1. Stock Maintenance System 5. Online course reservation system 2. E-ticketing 3. Software Personnel Management System
10	1. Credit Card Processing 2. E-book management System. 3. Recruitment system

1.	School offering the course	School of Computing
2.	Course Code	CSF302
3.	Course Title	Design and Analysis of Algorithms
4.	Credits (L:T:P:C)	3:0:1:4
5.	Contact Hours (L:T:P)	3:0:2
6.	Prerequisites (if any)	CSF102
7.	Course Basket	Discipline Core

COURSE SUMMARY

This course gives comprehensive introduction of computer algorithms with their time and space complexity. It provides example algorithms understanding of various categories like Divide & Conquer, Greedy, Dynamic Programming, Backtracking, and Branch & Bound. It introduces the problems that comes under category of P and NP.

COURSE OBJECTIVES

This course aims to provide the knowledge and understanding the various fundamental and advance data structures with their operational algorithms and complexity issues of algorithms. It aims to develop the ability to create algorithms for any task with best complexity.

COURSE OUTCOMES

After the study of this course student will be able to:

CO1. Understand and apply new algorithms.

CO2. Analyze complexity issues of algorithms

CO3. Create appropriate algorithm for performing any task.

CO4. Understand the existing and new algorithms in terms of P and NP Class problems.

CURRICULUM CONTENT

Unit-I **(6 L)**

Introduction: Algorithms, Performance Analysis: Space and Time Complexity, Asymptotic Notations- Big Oh, Omega, theta notations, finding complexity of the algorithm, Sorting: Insertion sort, Bubble sort, selection sort, count sort.

Unit –II **(8 L)**

Recurrence relation and solution (substitution, recurrence tree and master method).

Divide and Conquer: General method, binary search, quick sort, merge sort, heap sort

Unit –III **(8 L)**

Greedy Method: General method, Activity Selection, job scheduling with deadlines, fractional knapsackproblem, Minimum cost spanning tree: Kruskal‘s and Prim‘s, single source shortest path, Huffman tree.

Unit – IV **(9 L)**

Dynamic Programming: General Method, 0-1 Knapsack, Matrix chain multiplication, longest subsequence, all pair shortest paths,

Backtracking: Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of subsets.

Unit –V **(6 L)**

Branch and Bound: Travelling Salesman Problem

NP-Hard and NP-Complete problems: Basic Concepts, non-deterministic algorithms, NP-Hard and NP-Complete classes.

TEXT BOOKS:

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, ―Introduction to Algorithms‖, MIT Press;3rd edition, 2010.
2. Ellis Horowitz, SatrajSahni and Rajasekharam, Fundamentals of Computer Algorithms, UniversitiesPress; Second edition, 2008.

REFERENCE BOOKS:

1. R.C.T.Lee, S.S.Tseng, R.C.Chang and T.Tsai, Introduction to Design and Analysis of Algorithms Astrategic approach, McGraw-Hill Education (Asia), 2012.
2. Aho, Ullman and Hopcroft ,Design and Analysis of algorithms, Pearson Education India; 1st edition2010
3. Anany Levitin, Introduction to the Design and Analysis of Algorithm‖, Pearson Education India;2nd edition, 2008.

1.	School offering the course	School of Computing
2.	Course Code	**CSF303**
3.	Course Title	**Computer Networks**
4.	Credits (L:T:P:C)	3:0:1:4
5.	Contact Hours (L:T:P)	3:0:2
6.	Prerequisites (if any)	CSF201
7.	Course Basket	Discipline Core

COURSE SUMMARY

The course is a foundation level course and provides an in-depth description of computer networks. It begins by introducing the fundamentals of data communication and proceeds through the protocol layering architecture. It covers the physical layer by introducing the conversion of the analog and digital signals, transmission impairments, and transmission media. It also includes the data link layer and its services through protocols, network layer, IP address, delivery & forwarding packets, and network-layer protocols. Finally, it describes the transport layer & application layer that includes flow control, error control, congestion control, and application layer protocols like HTTP, FTP, SMTP, etc.

COURSE OBJECTIVES

The main objective of this course is to introduce you the fundamental concept of computer networks, how to build a network, what are the software & hardware requirements, how to analyze a network for performance and quality of service, and how two computers connected to a network communicate with each other.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Develop an ability to describe what a computer network is and how data communication takes place between two computers connected to a network.

CO2: Understand the protocol layering architecture and the different functions of each layer.

CO3: Explain the IPV4 addressing technique, including classful & classless address along with subnetting.

CO4: Develop an ability to analyze a network for their performance, quality of service, and throughput.

CURRICULUM CONTENT

Unit 1: Introduction to Computer Networks (8 L)

Data Communication and Network Fundamentals: Components of a Data Communication System, Data Flow, Computer Network and Internet, Network Topology, Network Models, Network Protocols, The Internet, History of Computer Network and the Internet.

Network Model and Layering Architecture: Network core: Packet Switch and Circuit Switch Network, A Network of Networks, Delay, Loss, and Throughput in Packet-Switched Networks, Protocol Layer and their Service Model: Layered Architecture, OSI and TCP/IP model.

Unit-2: Physical and Data Link Layer (7 L)

Physical Layer: Theoretical basis for communication, guided transmission media, wireless transmission, the public switched telephone networks, mobile telephone system, multiplexing: FDM, WDM, TDM, Transmission Media: Guided and Unguided Media,

Data Link Layer: Introduction to Data Link Layer, Services provided by the Data Link Layer, Error Detection and Correction Techniques, CRC, Checksum, Media Access Control: Random access protocol, Controlled Access Protocol, Ethernet and Ethernet Protocol.

Unit 3: Network Layer **(8 L)**

Introduction to Network Layer, Packet switching at network layer, Network Layer Services, Logical Addressing, IPV4 addresses: classful and classless, Subnetting, Delivery and Forwarding of Packets: Direct Delivery, Indirect Delivery, Next-hop method, Network Specific Method, Host Specific Method, forwarding with classful and classless addressing, Routing, Structure of a Router and switching techniques, Network Layer Protocols like ARP, RARP, ICMP etc. Unicast Routing Protocol: RIP, OSPF, BGP , Multicast Routing Protocol.

Unit 4: Transport Layer **(7 L)**

Introduction and Transport Layer Services: Process-to-Process Communication, Encapsulation and Decapsulation, Multiplexing and Demultiplexing, Flow Control, Error Control, Congestion Control, Connection-less and Connection-oriented services, Transport Layer Protocol: Simple protocol, Stopand-wait protocol, Go-back-N protocol, Selective-repeat protocol, TCP and UDP.

Unit 5: Application Layer **(8 L)**

Introduction to Application Layer, Application Architecture: Client-Sever, Peer-to-Peer, Process Communication, Client-Server communication Interface: Socket, IP, Using the services of Transport Layer, Application Layer Protocols: HTTP, FTP, SMTP, POP, IMAP, DNS.

TEXTBOOK(S)

1. Behrouz Forouzan, Data Communications, and Networking; McGraw Hill Education; 4th Edition (2017).
2. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach Pearson Education; Sixth edition (2017)

REFERENCE BOOKS

1. Andrews S. Tanenbaum, David J Wetherall; Computer Networks; Pearson Education; 5th Edition, 2013
2. Peterson, Larry L., and Bruce S. Davie. *Computer networks: a systems approach*. Elsevier, 2007.

List of Experiments

S.NO.	EXPERIMENT NAME
1	Simulate a network having two communication node using Cisco packet Tracer.
2	Simulate a network having 4 communication nodes with one switch.
3	Simulate a network having Two subnet using 2 switch, one Router and 6 nodes using Cisco packet tracer
4	Simulate a network using Star Topology Using Cisco packet Tracer.
5	Simulate a network using Bus Topology Using Cisco packet Tracer.
6	Simulate a network using Ring Topology Using Cisco packet Tracer.
7	Simulate a network using Mesh Topology Using Cisco packet Trace.
8	Create a DHCP server using Cisco packet tracer
9	Implement Intra domain and Inter domain routing Protocol using Cisco Packet Tracer.
10	Implement of Bit Stuffing and CRC.

1.	School offering the course	School of Computing
2.	Course Code	**CSF304**
3.	Course Title	**Artificial Intelligence**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF102
7.	Course Basket	Discipline Core

COURSE SUMMARY

The course will start with a brief introduction to artificial Intelligence. This course includes basic AI search techniques like A*, BFS, DFS. Introduction to Prolog is also important part of the content. Knowledge Representation, Reasoning Planning and Learning being requirement for development of expert system is also part of this course.

COURSE OBJECTIVES

The course is proposed to teach concepts of Artificial Intelligence. The subject will provide the foundations for AI problem solving techniques and knowledge representation formalisms.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Identify and formulate appropriate AI methods for solving a problem.
CO2: Apply AI algorithms.
CO3: Compare different AI algorithms in terms of design issues, computational complexity, and assumptions.
CO4: Utilize the concepts of AI for real world problem solving.

CURRICULUM CONTENT

Unit I: Introduction (6 L)

Introduction- Definitions, History, Characteristics, Applications, Intelligent Agents, Agent Environment, Types of Intelligent Agents, Environment Types. Introduction to python, Basic syntax, Basic operations, Loops, Data types, Functions.

Unit II: Problem solving and search (9 L)

Problem solving techniques, Search Terminologies, Properties of Search Algorithms, Search Algorithms- Uninformed Search, Informed Search, Minimax Search, Constraint satisfaction problem.

Unit III: Knowledge Representation (7 L)

Knowledge Representation- Introduction, Approaches and Issues in Knowledge Representation, Propositional Logic and Inference, First-Order Logic and Inference, Unification and Resolution, Expert Systems.

Unit IV: Reasoning (8 L)

Reasoning- Introduction, Types of Reasoning, Probabilistic Reasoning, Probabilistic Graphical Models, Certainty factors and Rule Based Systems, Introduction to Fuzzy Reasoning.

Unit V: Planning and Learning (7 L)

Planning and Learning- Introduction to Planning, Types-Conditional, Continuous, Multi-Agent. Introduction to Learning, Categories of Learning, Inductive Learning, Supervised and Unsupervised & Reinforcement Learning, Basic Introduction to Neural Net Learning.

TEXT BOOKS:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Pearson Education India; 3rd edition (2015)

2. Elaine Rich, Kevin Knight and Shivashankar B. Nair, "Artificial Intelligence", McGraw-Hill Education; 3rd edition (2017).

3. Nils J. Nilsson, "Artificial Intelligence-A New Synthesis", Morgan Kaufmann Publishers, Inc.; 1st edition (1998).

REFERENCES:

1. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Pearson Education India; 1st edition (2015)

List of Experiments

S.NO.	EXPERIMENT NAME
1	Write a program to solve water jug problem.
2	Write a program to implement Breadth-first Search algorithm.
3	Write a program to implement Depth-first Search algorithm.
4	Write a program to implement Best-first Search Algorithm.
5	Write a program to implement A* Search Algorithm
6	Write a program to print the root node using mini max algorithm.
7	Write a program to implement constraint Satisfaction problem.

1. School offering the course	School of Computing
2. Course Code	**CSF305**
3. Course Title	**Theory of Computation**
4. Credits (L:T:P:C)	3:1:0:4
5. Contact Hours (L:T:P)	3:1:0
6. Prerequisites (if any)	CSF202
7. Course Basket	Discipline Core

Course Summary

The course introduces fundamental concepts in the theory of computations and formal languages. This course contains types of languages and related grammars. This includes the detailed concepts of finite automaton, regular expression, context free grammars, pushdown automaton and Turing machines. It also includes introductory concepts of its applications into other area of computer science.

Course Objectives

This course will facilitate the students to learn the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability. In this course students will able to learn formalization of the notions via formal languages. The perspective learners will able to understand the hierarchy of classes of problems or formal languages.

Course Outcomes

On successful completion of the course, students will be able to achieve the following:
CO1. Demonstrate the basic elements of computation and the knowledge of finite automata.
CO2. Construct the grammars corresponding to the learned automata.
CO3. Analyze and able to construct the pushdown automata & Turing machine for the application problems.
CO4. Predict the decision problems and learn about the undecidable problems.

CURRICULUM CONTENT

Unit 1: Introduction to Finite Automata. (8 L)

Introduction to Mathematical foundation for automata: Mathematical preliminaries, alphabets, strings, languages, states, transition, transition graph, generalized transition graph.

Finite Automata: Deterministic Finite Automata, Non-Deterministic Finite Automata, Non-Deterministic Finite Automata with ϵ transitions, minimization of DFA.

Unit 2: NFA & FA with output (7 L)

Conversions and Equivalence: Equivalence between NFA with and without ϵ transitions. NFA to DFA conversion.

Application of FA: Equivalence between two DFA's, Limitations of FSM; Application of finite automata, Finite Automata with output- Moore & Melay machine and its conversion.

Unit 3 Grammars & context Free Language (8 L)

Regular Languages: Regular sets; Regular expressions, Arden's theorem, Construction of finite Automata for a given regular expression, Pumping lemma for regular sets. Closure properties of regular sets. Grammar Formalism: right linear and left linear grammars; Equivalence between regular linear grammar and FA.

Context free grammar: Grammar for CFL, Derivation trees, sentential forms. Ambiguity in context free grammars; Normal forms: Chomsky normal form and Greibach normal form; Pumping Lemma for Context Free Languages, Closure property of CFL.

Unit-4 Pushdown Automata (8 L)

Push Down Automata: Push down automata, definition; Acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence; Equivalence of CFL and PDA; Introduction to DCFL and DPDA

Unit- 5: Turing Machine & Computational Decidability (8 L)

Turing Machine: Turing Machine, definition, model, Design of TM, Computable functions Church's hypothesis, Types of Turing machines, Universal Turing Machine, Halting problem.

Properties and Decision problems: Properties of recursive and recursively enumerable languages, unsolvable decision problem, undecidability of Post correspondence problem, Church Turing Thesis.

TEXT BOOKS:

1. Hopcroft H.E. and Ullman J. D ,"Introduction to Automata Theory Language and Computation",., Pearson Education.3rd Edition.2008.
2. J. C. Martin, "Introduction to Languages and the Theory of Computation ", 3rd edition, Tata McGraw-Hill.2009.
3. K.L.P. Mishra, "Theory of Computer Science", PHI.3rd Edition 2014.

REFERENCES:

1. Lewis H.P. & Papadimitrou "Elements of Theory of Computation", C.H. Pearson, PHI.2nd Edition 2011.
2. Michael Sipser "Introduction to the Theory of Computation", Thomson India 2nd Edition(international) 2004.

1.	School offering the course	School of Computing
2.	Course Code	**CSF401**
3.	Course Title	**Compiler Design**
4.	Credits (L: T:P:C)	3:0:1:4
5.	Contact Hours (L: T:P)	3:0:2
6.	Prerequisites (if any)	CSF202
7.	Course Basket	Discipline Core

COURSE SUMMARY

The course is proposed to teach the students basic techniques that are used for the compiler design. The course introduction to all the phases of compiler and will introduce the theory and tools that are standardly employed in order to perform conversion of a high-level programming language into machine level code.

COURSE OBJECTIVES

The main objective of this course in to learn about different phases of compiler design and various types of grammars used in compiler design with practical exposure.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Identify various types' tokens from high level language program.
CO2: Analyze various semantic rules and its importance and apply different parsing techniques.
CO3: Create symbol table and understand fundamental of runtime environment.
CO4: Apply code optimization techniques and error handling techniques

CURRICULUM CONTENT

Unit 1: Introduction (7 L)

Compiler and Interpreter- Basic Concepts. Phases and Passes, Design Issues using Finite State Machines, Review of Languages & Grammar, Derivation and Parse Trees, Capabilities of CFG, Ambiguous Grammar, BNF Notation.

Unit 2: Basic Parsing Techniques (8 L)

Parsing-Top Down and Bottom-Up Strategies: General Consideration. Top Down Parsing: Brute-Force Method, Recursive Descent, & Predictive Paring. Bottom-Up Parsing: Shift Reduce Parsing, Operator Precedence Parsing. LR Grammars-LR(0), SLR(1), Canonical LR(1) & LALR(1) Parser, Comparison of parsing methods.

Unit 3 Semantic Analysis (8 L)

Basic Concepts, Syntax Directed Definitions-Inherited & Synthesized Attributes, Evaluation Orders of SDDs. Syntax directed Translation Schemes, Intermediate Codes, Postfix notation, Parse Trees and Syntax Trees, Directed Acyclic Graphs, Three address Codes: Quadruple & Triples, Translation of Assignment Statements, Boolean expressions, Control Statements, Postfix Translation, Translation with a Top Down Parser, Array References in Arithmetic expressions, Procedure Calls, Declarations and Case Statements Translations.

Unit-4 Symbol Tables (8 L)

Organization of Non-Block Structured Language (Unordered /Ordered/ Tree/ Hash) and Block Structured Language (Stack Tables & Stack Implementation), Runtime Storage Management: Static Allocation, Dynamic Allocation- Activation Records and their usage, Recursive Procedure. Heap Allocation-Storage Registers and Release Strategies.

Unit- 5: Error detection and Recovery **(7 L)**

Code Optimization- Basic Blocks and Optimization, Loop Optimization, Flow Graph Analysis, Machine Dependent Optimization. Error Handling: Detection, Reporting, Recovery and Maintenance, Compiler-Compiler—YACC, Code Generation, Concept of Compiler Design for Object-Oriented Language.

TEXT BOOKS:

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Compilers-Principles, Techniques & Tools, Pearson Education, 2nd Edition, 2018

REFERENCES:

1. Robin Hunter, Essence of Compilers, Pearson Education, 2004.
2. Steven S. Muchnick, Advanced Compiler Design & Implementation, Morgan Kaufmann Publishers, 2nd Edition, 2007.

List of Experiments

S.NO.	EXPERIMENT NAME
1	Write a program in C to count number of spaces in a line.
2	Write a program in C to count number characters, digits and spaces in a line.
3	Write a C program to recognize strings under 'a*' and 'abb'
4	Write a C program to test whether a given identifier is valid or not. (Logical, arithmetic)
5	Write a program using Lex to print any arithmetic expression in the form of tokens E.g. 2 + 4 * 3 O/p – Number plus/op Number Multi/op Number
6	Write a program in Lex to identify whether letter is consonant or vowel. E.g. gari O/P – consonant vowel consonant vowel
7	Write a C program to test whether a given identifier is valid or not. Apply regular expression l(l+d)*, where l- letters (a-z , A-Z) and d- digits (0 to 9).
8	Install Flex for windows. Write a program to print whether the word is a collection of lowercase or upper case. Also, print the total number of upper and lower letters
9	Write a program using Lex to print any arithmetic expression (+, -, *, /, %) in the form of tokens
10	Write a program using Lex to print to whether the number is even or odd

Artificial Intelligence, Machine Learning and Robotics

S. No.	Course Code	Course Title	Credits: L T P C
1.	CSF341	R Programming	2 0 2 3
2.	CSF342	Fuzzy Logic and Neural Network	2 0 2 3
3.	CSF343	Evolutionary Computing	2 0 2 3
4	CSF344	Machine Learning	2 0 2 3
5.	CSF441	Deep Learning	2 0 2 3
6.	CSF442	Robotics	2 0 2 3

1.	School offering the course	School of Computing
2.	Course Code	**CSF341**
3.	Course Title	**R Programming**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

In this course student will learn how to program in R and how to use R for effective data analysis. The course includes the installation and configuration of R programming a statistical programming environment, discuss generic programming language concepts and R data objects as they are implemented in a high-level statistical language. The course covers practical issues in statistical computing which includes programming in R, reading data into R, accessing R packages, writing R functions for f and Machine Learning algorithms.

COURSE OBJECTIVES

The objective of this course is to develop a broad perspective about the R programming and its applications to solve basic mathematical problems, statistical manipulations and scientific tasks such as data science and machine learning. R programming has its own built in functions to perform any specialized task. The course is intended to learn the basics of R software in this course.

COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1: Apply the basic functionalities of R programming to solve basic mathematical problems.

CO2: Apply the R programming for preprocessing the real-life datasets.

CO3: Understand and analyze the descriptive statistics for a given dataset.

CO4: Implement some classical machine learning models using R programming.

CURRICULUM CONTENT

UNIT 1 **(5L)**

What is R?, What is S? Basic Features of R, Limitations of R, R Framework setup, R packages, Use R like calculator, Reading and Writing data into R: combine or concatenate command, scan command, alternative commands for reading data, R constant and variables, operators and expression.

UNIT 2 **(5L)**

R data types and objects: Number and Text, Vector, Matrix, Factor, Array, List Data Frame, Manipulating Objects. Control structures, looping, scoping rules, Operations on Dates and Times, functions, debugging tools. R built-in packages and functions.

UNIT 3 **(5L)**

Dataset: Import/export bigger datafile (csv, text, excel, table, url, etc.), Identify and handle missing values, data formatting, Data Standardization, Data Normalization and Scaling, Data visualization, Binning, Multimedia datasets: text dataset, image dataset, audio dataset, video dataset.

UNIT 4 **(5L)**

Central tendency, Dispersion variance, standard deviation, shape skewness, kurtosis, percentiles, five-point summary, boxplots, histograms, bar plot, pie chart, scatter plot, two-way tables, covariance, correlation, Chi-

Square test for two-way tables.

UNIT 5 **(6L)**

Introduction to machine learning, types of machine Learning, supervised learning using R- regression, decision tree, KNN, SVM, Unsupervised learning using R- Clustering: K-means, hierarchical, frequent itemset, dimensionality reduction.

TEXTBOOK(S)

1. R programming for data science. R. D. Peng, Leanpub, 2016.
2. Practical Data Science with R. Author(s): Nina Zumel, John Mount, Manning Shelter Island, 2014.

REFERENCES :

1. The R book, Crawley and Michael, John Wiley & Sons, 2012.
2. Beginning R: The statistical programming language. Mark Gardener, John Wiley & Sons, 2012.

1.	School offering the course	School of Computing
2.	Course Code	**CSF342**
3.	Course Title	**Fuzzy Logic and Neural Network**
4.	Credits (L: T:P:C)	2: 0: 1 :3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	None
7.	Course Basket	DE

COURSE SUMMARY

The course is proposed to teach students the concepts of Fuzzy Logic and Neural Networks. Students will develop understanding of the different neural network algorithms and fuzzy functions.

COURSE OBJECTIVES

The course will facilitate the students to learn the fundamentals of Fuzzy Logic and Neural Networks.

COURSE OUTCOMES

Course Outcomes (COs): After the completion of the course, students will be able to:

CO1. Understanding the concepts of Fuzzy Logic.
CO2. Applying different fuzzy operations and functions.
CO3. Understanding the concepts of Neural Network.
CO4. Understanding and implement different Activation Functions.

CURRICULUM CONTENT

UNIT 1 **(6L)**

Introduction to Soft Computing: Concept of computing systems, "Soft" computing versus "Hard" computing, Characteristics of Soft computing. Fuzzy Computing, Neural Computing, Applications of Soft computing techniques.

UNIT 2 **(7L)**

Fundamentals of Fuzzy Logic: Basic Concepts: Fuzzy Set Theory, Basic Concepts of Crisp sets and fuzzy set, complements, union, intersection, combination of operations, general aggregation operation, fuzzy relations, fuzzy proposition, fuzzy implication, compatibility relation. Fuzzy membership function, Defuzzification Techniques.

UNIT 3 **(7L)**

Introduction to Neural Networks: Introduction to Biological Neural Network, Artificial Neural Network. Activation Functions, Basic Learning Rules, Hebb's rule, Biases and Threshold, Perceptron, Convergence Theorem, Delta Rule, Hyperparameter, Cost Function, Applications of Artificial Neural Networks.

UNIT 4 **(6L)**

Neural Network Techniques: Gradient Descent, Stochastic Gradient Descent, Back Propagation, Multi-Layer Perceptron, Feed Forward Networks, Convolution Neural Network, Recurrent Neural Networks, Bayesian Network, Hopfield Network, Radial Basis Network.

UNIT 5 **(7L)**

Advanced Neural Networks: Architecture of Cognitron and Neocognitron, Auto Encoders, Gated Recurrent Unit, Long Short-Term Memory, Kohonen Self Organizing Network, Modular Neural Network.

TEXTBOOK(S):

1. George J. Klir / Bo Yuan , Fuzzy Sets and Fuzzy Logic: Theory and A: Theory and Applications, Pearson Education India 2015
2. Laurene V. Fauset , Fundamentals of Neural Networks: Architectures, Algorithms and Applications, Pearson Education , 2006

REFERENCES:

1. Bart Kosko, ―Neural network and Fuzzy System‖ - Prentice Hall-1994.
2. J.Klin and T.A.Folger, ―Fuzzy sets‖ University and information- Prentice Hall -1996.
3. J.M.Zurada, ―Introduction to artificial neural systems‖-Jaico Publication house,Delhi 1994.
4. VallusuRao and HayagvnaRao , ―C++ Neural network and fuzzy logic‖-BPB and Publication, New Delhi,1996.

1.	School offering the course	School of Computing
2.	Course Code	**CSF343**
3.	Course Title	**Evolutionary Computing**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The evolution of computers has been phenomenal in the last decades with computers becoming part of each and every aspect of human lives. This course seeks to use the concepts of human evolution to become a part of the further evolution of computers. Using biological evolution as a motivation many computer problems can be solved much faster. This course seeks to guide students to how to implement and think these algorithms.

COURSE OBJECTIVES

The main goal of this course is to help students learn an evolutionary method for computer solvable problems. The course seeks to find out the solution for complex computing problems using Darwinian laws as its basic motivation in order to find better solutions to certain problems. Students shall be able to get familiar with advanced concepts of mutation and the implementation of these biological concepts through methods such as neural networks and statistical methods.

COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1: Understand the fundamental of evolution based learning algorithms, advanced searching and optimization techniques.

CO2: Analyze and Understand the concepts of genetic algorithms.

CO3: Ability to apply swarm intelligence and Ant Colony Optimization.

CO4: Ability to create algorithms evolutionary computing based algorithms for solving problem.

CURRICULUM CONTENT

UNIT 1 **(5L)**

Introduction, Optimization Problems, Problem Domains, Global Optimization and Techniques of Global Optimization: Branch and Bound, Clustering Methods, Hybrid Methods, Simulated Annealing, Statistical Global Optimization Algorithms, Taboo Search, Multi Objective Optimization, Darwinian Evolution, Genetics, What is an Evolutionary Algorithm, Components of Evolutionary Algorithms, Competitive Learning, Working of an Evolutionary Algorithm, Evolutionary Computing and Global Optimization.

UNIT 2 **(5L)**

Genetic Algorithm: Introduction, Representation of Individuals, Mutation, Recombination, Population Models, Parent Selection, Survivor Selection, Age-Based Replacement, Fitness Based Replacement, Evolutionary Strategies, Example Applications.

Genetic Programming: Introduction, Representation, Mutation, Recombination, Parent Selection, Survivor Selection, Initialization, Bloat in Genetic Programming, Problems Involving ``Physical" Environments, Example Applications.

UNIT 3 **(5L)**

Swarm Intelligence: Introduction, key principles (e.g., self-organization, stigmergy), natural and artificial examples, computational and real-time SI, Ant System (AS), the first combinatorial optimization algorithm based on ant trail/following principles, Travel Salesman Problem (TSP). Ant Colony Optimization (ACO), Ant-based algorithms (ABC, Ant-Net) applied to routing in telecommunication networks.

UNIT 4 **(5L)**

Multimodal problems, need for diversity, implicit measures, explicit diversity maintenance, multi objective evolutionary algorithms.

UNIT 5 **(6L)**

Evolutionary Robotics, Evolutionary Neural Networks, Dynamic Landscapes, Parallel EC, Multi objective EC.

TEXTBOOK(S)

1. A.E.Eiben & J.E.Smith. Introduction to Evolutionary Computing. Springer-Verlag Berlin Heidelberg, 2nd edition, 2016.

REFERENCES:

1. S. Sumathi &T.Hamsapriya & P.Surekha, Evolutionary Intelligence-An Introduction to theory and applications with Matlab Springer-Verlag Berlin Heidelberg, 3rdedision, 2008.
2. Kenneth A. De Jong, Evolutionary Computation, A unified Approach The MIT Press Cambridge, Massachusetts London, England, 1stedision, 2006.

Data Science and Analytics

S. No.	Course Code	Course Title	Credits: L T P C
1.	CSF341	R Programming	2 0 2 3
2.	CSF344	Machine Learning	2 0 2 3
3.	CSF345	Introduction to Data Science	2 0 2 3
4	CSF346	Data Mining and Data Warehousing	2 0 2 3
5.	CSF441	Deep Learning	2 0 2 3
6.	CSF443	Big Data Analytics	2 0 2 3

1.	School offering the course	School of Computing
2.	Course Code	**CSF344**
3.	Course Title	**Machine Learning**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

This course is designed to provide an introduction to techniques, and algorithms in machine learning, beginning with topics such as classification and linear regression and ending up with more recent topics such as support vector machines, decision tree, and Bayesian networks. The course will give the student the basic ideas and intuition behind modern machine learning methods as well as a bit more formal understanding of how, why, and when they work.

COURSE OBJECTIVES

This course is designed to provide knowledge about basic concepts of Machine Learning, identify machine learning techniques suitable for a given problem, solve the problems using various machine learning techniques, apply Dimensionality reduction techniques and design application using machine learning techniques.

COURSE OUTCOMES

After the completion of the course, students will be able to:

CO1: Analyse & Differentiate various learning approaches and to interpret the concepts of supervised and unsupervised learning.

CO2: Understand the different dimensionality reduction techniques.

CO3: Evaluate & illustrate the working of classifier models like SVM, Neural Networks and identify classifier model for typical machine learning applications.

CO4: Create & Apply clustering algorithms and identify its applicability in real life problems.

CURRICULUM CONTENT

UNIT 1 **(5L)**

Introduction: Probability Theory, Overview of machine learning: Unsupervised, Supervised, Reinforcement, Programs vs learning algorithms, goals & applications, software tools, machine learning problems, components of a learning, types of learning. Aspects of developing a learning system: training data, concept representation, function approximation.

UNIT 2 **(5L)**

Regression: Linear Regression, Ridge Regression, Sensitivity Analysis, Multivariate Regression.

Clustering: Distance measures, Different clustering methods (Distance, Density, Hierarchical), Iterative distance-based clustering, dealing with continuous, categorical values in K-Means, Constructing a hierarchical cluster, K-Medoids, k-Mode and density-based clustering, Measures of quality of clustering, Hidden Markov Model.

UNIT 3 **(5L)**

Classification: Bayesian Learning (Bayes theorem, Bayes Optimal Classifier, Naïve Bayes classifier),

K-Nearest Neighbors, Support Vector Machines, Decision Trees, Boosted Trees, Random Forest, CART, Gradient boosting.

UNIT 4 **(5L)**

Dimensionality Reduction: Feature selection, principal component analysis, linear discriminant analysis, factor analysis, independent component analysis, multidimensional scaling, manifold learning, band selection. Introduction to Analytical Learning, Combining Inductive and Analytical learning, Reinforcement learning, adaptive hierarchical clustering, Gaussian mixture model.

UNIT 5 **(6L)**

Artificial Neural Networks: The perceptron algorithm, multilayer perceptron, back propagation, Introduction to Deep Neural networks, Recurrent Neural Networks and Convolutional Neural Networks.

TEXTBOOK(S)

1. Tom Mitchell, Machine Learning, McGraw Hill, 1st edition, 2017.
2. Ethem Alpaydin, Introduction to Machine Learning, PHI, 3rd edition, 2015.

REFERENCES:

1. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2nd edition, 2013.
2. Stephen Marsland, Machine Learning: An Algorithmic Perspective, CRC Press, 2nd edition, 2014.

1.	School offering the course	School of Computing
2.	Course Code	**CSF442**
3.	Course Title	**Robotics**
4.	Credits (L: T:P:C)	2: 0: 1 :3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The course is proposed to teach the students the concepts of Robotics. Students will develop understanding of the different principles of sensors and methods of robot.

COURSE OBJECTIVES

The course will facilitate the students to learn the fundamentals, Techniques used in Robotics.

COURSE OUTCOMES

Course Outcomes (COs): After the completion of the course, students will be able to:

CO1. Understating of how to keep robots in modern industries.
CO2.. Applying robots in different areas (space, medical, manufacturing etc.).
CO3. Understating different components of robots system and their working principle.
CO4. Creating robot using robot lego robotics kit.

CURRICULUM CONTENT

UNIT 1 **(5L)**

Introduction to Soft Computing: Concept of computing systems, "Soft" computing versus "Hard" computing, Characteristics of Soft computing. Fuzzy Computing, Neural Computing, Applications of Soft computing techniques.

UNIT 2 **(5L)**

Actuators: Characteristics of Actuating Systems, Actuating Devices and Control. Sensors: Sensor Characteristics, Description of Different Sensors, Touch sensors, Tactile sensor, Proximity and range sensors, Robotic vision sensor, Force sensor, Light sensors, Pressure sensors.

UNIT 3 **(5L)**

Concepts of AI, AI Problems, techniques, Characteristics & Applications, AI versus Natural Intelligence, Problem representation in AI, Problem-solution Techniques. Elements of Knowledge Representation: Logic, Production Systems, Semantic Networks, Expert Systems. Defining the Problem as State Space Search, Production Systems, Production Systems, Issues in the Design of Search Programs, DFS & BFS Techniques

UNIT 4 **(7L)**

Introduction to lego robotics kits, Introduction to robot manipulation. Forward and inverse kinematics of robots and some case studies. Manipulator dynamics. Basics of robot control. Task planning with emphasis on computational geometry methods for robot path finding, robot arm reachability, grasp planning. Overview of robot vision and Parallel robots

UNIT 5 **(6L)**

Multi-robot representations and Task Planning: Task-Level Programming, Uncertainty, Configuration Space, Gross-Motion Planning, Grasp Planning, Fine Motion Planning, Task Planning Problem.: control architectures, simulation environments, and test beds. Integration of assorted sensors (IR, Potentiometer, strain gages etc.), micro controllers and ROS (Robot Operating System) in a robotic system.

TEXTBOOK(S)

1. Fundamentals of Robotics Analysis and Control, Robert J Schilling, PHI, 5thedision, 2012
2. Introduction to Robotics Analysis, Systems, Applications by Saeed B. Niku, Prentice Hall, 2nd Edition, 2014.

REFERENCES:

1. An Introduction to Multi Agent Systems, Michael Wooldridge Wiley, 2014
   J J Craig, "Introduction to Robotics: Mechanics

1.	School offering the course	School of Computing
2.	Course Code	**CSF345**
3.	Course Title	**Introduction to Data Science**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

This course aims to provide a basic understanding of Data Science concepts. This course introduces students to the data science principles required to tackle real-world, data-rich problems in business.

COURSE OBJECTIVES

Data Science is the study of the generalizable extraction of knowledge from data. This course serves as an introduction to the data science principles required to tackle real-world, data-rich problems in business and academia, including: Data acquisition, cleaning, and aggregation, Exploratory data analysis and Visualization, Feature engineering, Model creation and validation, Basic statistical and mathematical foundations for data science

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: An understanding of problems solvable with data science and an ability to attack them from a statistical perspective.

CO2: An understanding of when to use supervised and unsupervised statistical learning methods on labeled and unlabeled data-rich problems

CO3: The ability to create data analytical pipelines and applications in Python. .

CO4: Apply the various tools needed to continue developing as a data scientist.

.

CURRICULUM CONTENT

Unit 1: Computer Science/Statistics/Linear Algebra Short Review (5 L)

What is data science? Brief review of prerequisite knowledge for studying data science. Basics of computer science; data structures/types, program control flow, and syntax in Python. Basics of statistics; probability and probability distributions. Basics of linear algebra; matrices, vectors using Python programming language.

Unit 2: Exploratory Data Analysis (Eda) And Visualization Design (5 L)

E-R model: Basic concepts, Design Issues, Mapping Constraints, Attributes and Entity sets, Relationships and Relationship sets, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended ER features.

Unit 3 Data Modeling: Supervised/Unsupervised Learning (5 L)

Two basic kinds of statistical models used for prediction. Supervised Learning algorithm: Linear Regression and Logistic Regression. Unsupervised Learning algorithm: K-Means clustering. Advanced supervised learning algorithms like linear support vector machines, decision trees, and random forest models for regression and classification. Advanced unsupervised learning algorithm like DBSCAN.

Unit-4 Data Modeling: Feature Selection, Engineering, And Data Pipelines (5 L)

Curse of dimensionality and Dimensionality reduction. Feature selection and feature extraction. Principal Component Analysis/Independent Component Analysis and regularization. Construct complete data pipelines, going from data ingestion, preprocessing to model construction and evaluation.

Unit- 5: Data Modeling: Model Evaluation And Project Presentations (6 L)

Exploration of more sophisticated model evaluation approaches like cross-validation and bootstrapping with the goal of making the model as generalizable as possible. Presentation of students‘ project and sharing learning experience.

TEXT BOOKS:

1. Cathy O‘Neil and Rachel Schutt, Doing Data Science, Straight Talk from the Frontline, O‘Reilly. 2014.
2. Jiawei Han, Micheline Kamber and Jian Pei Silberschatz, Korth, Data Mining: Concepts and Techniques, Third Edition. ISBN 0123814790. 2011

REFERENCES:

1. Mohammed J. Zaki and Wagner Miera Jr, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press. 2014.

1.	School offering the course	School of Computing
2.	Course Code	**CSF346**
3.	Course Title	**Data Mining and Data Warehousing**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	Discipline Elective

COURSE SUMMARY

This course aims to provide a basic understanding of Data ware housing and mining concepts, implementation of Data Mining algorithms. This course introduces students to enterprise data and the process and technologies to integrate data from a variety of sources.

COURSE OBJECTIVES

This course will cover the basic concepts of Data Warehouse and Data Mining techniques, Examine the types of the data to be mined and apply pre-processing methods on raw data. It also discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Explain and evaluate the various data mining algorithms
CO2: Discover and measure interesting patterns from different kinds of databases.
CO3: Apply the techniques of clustering, classification, association finding,
CO4: Apply techniques for feature selection and visualization to real world data.

CURRICULUM CONTENT

Unit 1: Overview **(5 L)**

Motivation (for Data Mining), Data Mining-Definition & Functionalities. Data Warehousing: Overview, Definition, Delivery Process, Difference between Database System and Data Warehouse, Multi-Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept hierarchy, Process Architecture, 3 Tier Architecture, Data Marting. ROLAP, MOLAP, HOLAP.

Unit 2: Data Pre-Processing **(5 L)**

Data Cleaning: Missing Values, Noisy Data, (Binning, Clustering, Regression, Inconsistent Data, Data Integration and Transformation. Data Reduction: Data Cube Aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Clustering, Discretization and Concept hierarchy generation.

Unit 3 Concept Description **(5 L)**

Definition, Data Generalization, Analytical Characterization, Analysis of attribute relevance, Mining Class comparisons, Statistical measures in large Databases. Measuring Central Tendency, Measuring Dispersion of Data, Apriori Algorithm, Mining Multilevel Association rules from Transaction Databases

Unit-4 Classification **(5 L)**

What is Classification, Issues regarding Classification, Decision tree, Bayesian Classification, Classification by Back propagation.

Unit- 5: Cluster Analysis **(6 L)**

Data types in cluster analysis, Partitioning methods. Hierarchical Clustering- CURE and Chameleon, Density Based Methods-DBSCAN, OPTICS, Grid Based Methods STING, CLIQUE, Outlier Analysis

TEXT BOOKS:

1. Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques, Elsevier, Third Edition, 2012.

REFERENCES:

1. Margaret H. Dunham, Data-Mining: Introductory & Advanced Topics, Pearson Education, India, 3rd edition, 2012.

1.	School offering the course	School of Computing
2.	Course Code	**CSF441**
3.	Course Title	**Deep Learning**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	Discipline Elective

COURSE SUMMARY

This course aims to provide a basic understanding of deep learning concepts, implementation of supervised and unsupervised algorithms. This course introduces students to enterprise data and the process and technologies to integrate data from a variety of sources.

COURSE OBJECTIVES

The objective of this course is to cover the fundamentals of neural networks as well as some advanced topics such as recurrent neural networks, long short term memory cells and convolutional neural networks. The course also requires students to implement programming assignments related to these topics.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Understand the concept of artificial neural networks, convolutional neural networks, and recurrent neural networks

CO2: Discuss how to speed up neural networks along with regularization techniques to reduce overfitting.

CO3: Understand the concept of generative models.

CO4: Implement deep learning algorithms, and learn how to train deep networks.

CURRICULUM CONTENT

Unit 1: Basics (5 L)

Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic, linear Perceptron, Perceptron Learning Algorithm, Linear separability. Convergence theorem for perceptron Learning Algorithm.

Unit 2: Feed Forward Networks (5 L)

Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, autoencoders.

Unit 3: Feed Forward Networks (5 L)

Deep Neural Networks: Difficulty of training deep neural networks, Greedy layer wise training. Better Training of Neural Networks: Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem in neural networks, Regularization methods (dropout, drop connect, batch normalization).

Unit 4: Recurrent Neural Networks (5 L)

Recurrent Neural Networks: Back propagation through time, Long Short Term Memory, Gated Recurrent Units, Bidirectional LSTMs, Bidirectional RNNs , Convolutional Neural Networks: LeNet, AlexNet.

Unit 5: Generative Models **(6 L)**

Restrictive Boltzmann Machines (RBMs), Introduction to MCMC and Gibbs Sampling, gradient computations in RBMs, Deep Boltzmann Machines, Recent trends: Variational Autoencoders, Generative Adversarial Networks, Multitask Deep Learning, Multi-view Deep Learning, Applications: Vision, NLP, Speech and Deep Learning Tools.

TEXT BOOKS :

1. Ian Good fellow and Yoshua Bengio and Aaron Courville, Deep Learning, , MIT Press, 2016.

REFERENCES :

1. Raul Rojas, Neural Networks: A Systematic Introduction, Springer-Verlag, Berlin, New-York, 1996
2. Christopher Bishop, Pattern Recognition and Machine Learning, 2010

1.	School offering the course	School of Computing
2.	Course Code	**CSF443**
3.	Course Title	**Big Data Analytics**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	Discipline Elective

COURSE SUMMARY

Course Summary To learn the need for Big Data Analytics, and to acquire modern tools to implement in real life applications.

COURSE OBJECTIVES

Understanding the fundamentals of various big data analysis techniques, Hadoop structure, environment and framework.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the need and process of data analysis.
CO2: Learn the different component of Hadoop Ecosystem.
CO3: Design Map Reduce and the use of Apriori.
CO4: Apply and Analyse different software for processing Big Data

CURRICULUM CONTENT

UNIT 1: INTRODUCTION TO BIG DATA AND HADOOP (5L)

Types of Digital Data, Introduction to Big Data, Big Data Analytics, Analytic Processes and Tools, Analysis vs Reporting, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference, Prediction Error, Modern Data Analytic Tools - History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy.

UNIT 2: HADOOP DISTRIBUTED FILE SYSTEM (HDFS) (5L)

The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.

UNIT 3: MAP REDUCE (5L)

Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features. Mining Frequent Item sets :- Market Based Model, Apriori Algorithm, FP-Growth.

UNIT 4: HADOOP ECO SYSTEM **[5L]**

Pig: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Hive: Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions. Hbase: HBasics, Concepts, Clients, Example, Hbase Versus RDBMS. Big SQL: Introduction.

UNIT- 5:DATA ANALYTICS WITH R **[6L]**

Overview of R programming language, Regression Modelling, Multivariate Analysis. Machine Learning: Introduction, Supervised Learning, Unsupervised Learning, Collaborative Filtering. Big Data Analytics with BigR. Machine learning tools: Spark &SparkML, H2O, Azure ML.

TEXT BOOKS :

1. Michael Berthold, David J., Intelligent Data Analysis, 2/e, Springer, 2015.
2. Anand Raja Raman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.

REFERENCES :

1. Glenn J. Myatt Making Sense of Data, , John Wiley & Sons, 2014
2. Pete Warden , Big Data Glossary, O'Reilly, 2011.

Internet of Things and Edge Computing

S. No.	Course Title	Course Title	Credits: L T P C
1.	CSF347	Wireless and Mobile Systems	3 0 0 3
2.	CSF348	Mobile Application Programming using Android	2 0 2 3
3.	CSF349	Cloud Computing	2 0 2 3
4.	CSF351	Advanced Computer Networks	3 0 0 3
5.	CSF444	Internet of Things	2 0 2 3
6.	CSF445	Mobile & Wireless Network Security	2 0 2 3

1.	School offering the course	School of Computing
2.	**Course Code**	**CSF347**
3.	**Course Title**	**Wireless and Mobile Systems**
4.	Credits (L:T:P:C)	3:0:0:3
5.	Contact Hours (L:T:P)	3:0:0
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

Course Summary

This course deals with the basics of cellular concept and mobile communication systems, multiple radio access procedures and channel allocation techniques, the architecture and functioning of satellite systems including global positioning systems, different wireless LAN technologies and personal area networks.

Course Objectives

This course aims to provide students a comprehensive overview of different types of wireless and mobile systems with a detailed focus on architecture of modern-day cellular systems. Students will learn concepts about mobile communication systems architecture, wireless standards, satellite systems as well as personal area networks.

Course Outcomes

On successful completion of the course, students will be able to achieve the following:

CO1: Understand various radio propagation mechanisms

CO2: Understand cellular concepts, multiple division techniques and channel allocation techniques.

CO3: Understand Mobile Communication System Architecture

CO4: Understand Wireless MANS, LANS and PANS.

Curriculum Content

Unit 1: History of wireless systems. Introduction to various types of wireless and mobile systems. **(3 L)**

Unit 2: Types of Radio Waves, Propagation Mechanisms, Free Space Propagation, Land Propagation, Path loss and Fading, Doppler Effect, Delay Spread and Intersymbol Interference. **(6 L)**

Unit 3: Cellular Concept, Cell Area, Signal Strength and Cell parameters, Capacity of a cell, Frequency reuse, How to form a cluster, Cochannel Interference, Cell Splitting and Cell Sectoring, Multiple division Techniques, Concepts and Models of Multiple Divisions (FDMA, TDMA, etc.), Channel Allocation, Static Allocation versus Dynamic Allocation, Fixed Channel Allocation, Dynamic Channel Allocation, Hybrid Channel Allocation, Allocation in specialized System Structure. **(12L)**

Unit 4: Mobile Communication Systems, Cellular System Infrastructure, Registration, Handoff and Roaming Support, Multicasting, Security and Privacy. **(6 L)**

Unit 5: Wireless MANs, LANs and PANs, Wireless Metropolitan Area Networks (4G systems), Wireless Local Area Networks (IEEE 802.11x), Wireless Personal Area Network (Bluetooth Networks), Case Studies of all these types of networks. **(12L)**

Textbook(s)

1. Introduction to Wireless and Mobile Systems, by D.P. Agrawal and Q. Zeng, Cengage, 3rd Edition, 2012.

Reference Books

1.Wireless Communications and Networking, V.K.Garg, Morgan Kaufmann, 1st Edition, 2008

1.	School offering the course	School of Computing
2.	**Course Code**	**CSF348**
3.	**Course Title**	**Mobile Application programming using Android**
4.	Credits (L:T:P:C)	2:0:1:3
5.	Contact Hours (L:T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

Course Summary

This course deals with the internals of Android Operating System, GUI, various services, graphics design, database connectivity, network connectivity and integration of various APIs.

Course Objectives

The objective of this course is to teach mobile application programming to students using Android. Students will learn about the technologies and the tools used to develop Android mobile applications. Students will be introduced to the internals of the Android OS and mobile application development using the Android SDK.

Course Outcomes

On successful completion of the course, students will be able to achieve the following:

CO1: Understand internals of the Android OS
CO2: Implement mobile application development using the Android SDK.
CO3: Implement GUI, Services, Database Connectivity and Web Service Integration
CO4: Understand network connectivity and integration of various APIs.

Curriculum Content

Unit 1: Overview of mobile applications, installing the development environment, Android Overview, architecture overview and Android development environment, Anatomy of an Android App, App lifecycle. **(3L)**

Unit 2: GUI development: XML for UI design, development tools, Activities, multiple activities, Activity lifecycle, Intents, MVC, GUI development, Lists, fragments, dialogs, Action Bar, 2D graphics and drawables. **(12L)**

Unit 3: Services and Broadcast Services, Database connectivity with SQLite, Web service integration using JSON, XML, SOAP and RESTful services. **(6L)**

Unit 4: Network connectivity, Integration with multiple APIs. **(3L)**

Textbook(s)

1. Head First Android Development: A Brain-Friendly Guide, by Dawn Griffiths and David Griffiths.

Reference Books

1. Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides), 2017.

1.	School offering the course	School of Computing
2.	Course Code	**CSF351**
3.	Course Title	**Advanced Computer Networks**
4.	Credits (L:T:P:C)	3:0:0:3
5.	Contact Hours (L:T:P)	3:0:0
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

Course Summary

The course is designed for understanding of various routing proptocols. The course introduces the queuing model and Markovian theory to handle process state at the various stages of the switches and routers. The course introduces the basic Wireless security to understand the various network's attack and prevention. A number of various wireless standards are also included in this course.

Course Objectives

This course is designed to provide knowledge about some of the advanced concepts of Computer Network like network routing design, wireless LAN standards, stochastic processes and queueing concepts, and network security.

Course Outcomes

On successful completion of the course, students will be able to achieve the following:

CO1: Understand and explain the concepts of network routing.

CO2: Understand various Wireless LAN standards.

CO3: Understand stochastic processes and queueing systems.

CO4: Understand Network Security and Management Design Techniques

Curriculum Content

Unit 1: Switching and routing

Routing Concepts & operations. Dynamic Routing Protocols, Distance Vector Routing Protocols, RIP(IPv4) and RIPng(IPv6) Routing, Link-State Dynamic Routing, The Routing Table, Single-Area OSPF, Configuring Single-Area OSPFv2 (IPv4) & v3(IPv6). **(9L)**

Unit 2: Wireless LANs

Wireless Technology Overview, Wireless Standards, Wireless Components, Wireless Security, Wireless Design Considerations, IEEE 802.11 standards, Cellular Networks, Mobile IP, Wireless Mesh Networks (WMNs). **(9L)**

Unit 3: Stochastic Processes & Queuing Systems

Stochastic Processes: The Poisson Process, Birth Death Process, Markov Chains. Single Station Queuing System: Kendall‘s Notation, Performance Measures, The M/M/1 Queue, The M/M/$\propto$ Queue, The M/M/m Queue. **(9L)**

Unit 4: Network Security and Management Design

Hacking: Vulnerabilities, Threats: Reconnaissance Attacks, Access Attacks, Information Disclosure Attacks, Denial of Service Attacks, Threat Defence Secure Communication, Network Security Best Practices, SAFE Campus Design. ISO Network Management Standard: Protocols and Tools, SNMP, MIB, RMON, Cisco NetFlow, Syslog, Network Management Strategy: SLCs and SLAs, IP Service Level Agreements, Content Networking Design. **(6L)**

Textbook(s)

1. Network Routing: Algorithms, Protocols, and Architectures, Deep Medhi, K. Ramaswamy, Morgan Kaufmann, 2nd Edition, 2017.
2. Probability & Statistics with Reliability Queuing and Computer Science Applications, Kishore S Trivedi, 2Wiley, nd Editon, 2008.

1.	School offering the course	School of Computing
2.	Course Code	**CSF444**
3.	Course Title	**Internet of Things**
4.	Credits (L:T:P:C)	2:0:1:3
5.	Contact Hours (L:T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

Course Summary

In this course students will be introduced to fundamental and architectural concepts of IoT systems, various kinds of communication using system-on-chip devices and building IoT prototypes. Students will learn how to create an end-to-end system by connecting to IoT cloud, perform IoT Analytics and understand cloud security.

Course Objectives

The objective of this course is to provide both conceptual and hands-on knowledge to students for IoT systems. Students will learn how to build and use end-to-end IoT systems, perform analytics on the data collected and understand security aspects of an IoT system.

Course Outcomes:

On successful completion of the course, students will be able to achieve the following:

CO1: Understand fundamental concepts and building blocks of an IoT system.
CO2: Understand and implement IoT prototypes using system-on-chip devices.
CO3: Understand and develop end-to-end systems by syncing with Cloud.
CO4: Understand security aspects of an IoT system.

Curriculum Content

Unit 1: Introduction, IoT Architecture, Sensing, Communication and Actuation, Hardware and Software setup **(3L)**

Unit 2: GPIO pins setup and programming, Serial Communication in IoT, SPI and I2C in IoT. **(15L)**

Unit 3: Data transmission in Cloud, IoT Analytics and Visualization **(3L)**

Unit 4: IoT Security, IoT Project execution and demonstration **(3L)**

12. Bibliography

Textbook: No Textbook. Instructors will provide reading materials.

Cyber Security and Privacy

S. No.	Course Code	Course Title (Proposed)	Credits: L T P C
1.	CSF352	Number Theory and Cryptology	2 0 1 3
2.	CSF353	Foundation of Cyber Security	2 1 0 3
3.	CSF354	Data Encryption & Network Security	2 0 2 3
4	CSF355	Cyber Crime & Investigation	2 0 2 3
5.	CSF445	Mobile & Wireless Network Security	2 0 2 3
6.	CSF446	Ethical Hacking & Digital Forensics	2 0 2 3

1. School offering the course	School of Computing
2.Course Code	**CSF352**
3.Course Title	**Number Theory and Cryptology**
4.Credits (L:T:P:C)	2:0:1:3
5.Contact Hours (L:T:P)	2:0:2
6.Prerequisites (if any)	NA
7.Course Basket	DE

COURSE OUTLINE:

This course will introduce the basic concepts of cryptography, which includes the Substitution & Transposition Techniques, Public Key and Secret Key Cryptography. The course will consist of assigned reading, weekly lectures, weekly practical, a midterm and final exam, and a sequence of class test and assignments. The goal of the readings and lectures is to introduce the core concepts. The goal of the practical is to give students some exposure to secure code designing.

COURSE OBJECTIVE:

To introduce the student to elementary number theory, as required for further study of important cryptographic protocols. To introduce the student to the fundamentals of modern symmetric cryptography. To enable the student to appreciate the significance of cryptography as a means of securing information in the modern world.

COURSE OUTCOMES:

At the end of the course, the student will be able to:
CO 1. Understand the significance of cryptography to the modern world and the internet.
CO 2. Understand the rationale behind block cipher designs.
CO 3. Utilize the cryptanalysis of a simple block ciphers.
CO 4. Solve elementary problems in number theory relating to cryptography.

CURRICULUM CONTENT

UNIT 1: (6 L)

Basic Cryptography Concepts- Basic Cryptography Concepts, Purpose of Cryptography Need for security, Security Goals, Principles of security,Types of attacks.

Encryption Techniques: Plaintext, Cipher text, Substitution & Transposition Techniques,

Classical methods: Caesar cipher, Vigenere cipher, The one-time pad, Mechanical rotor systems, Vernam Cipher, Affine Cipher, Hill Cipher, Playfair Cipher, Rail Fence Cipher, Columnar Cipher

UNIT 2: (6 L)

Modern ciphers: Block ciphers and their applications, Structure of a block cipher, The Fiestel structure, Key and block size length, The Data Encryption Standard (DES), Double DES, Triple DES, AES.

UNIT 3: (5 L)

Elementary Number Theory: Finite fields, Modular arithmetic, Efficient algorithms for modular arithmetic, Fermat's little theorem, Euler's criteria, Euler's totient function.

UNIT-4 (5L)

Advanced Number Theory: Primality testing, prime factorisation, The Chinese remainder theorem, Quadratic residues and calculating modular square roots and cube roots.

UNIT- 5: **(5 L)**

Public Key Cryptography &Key Distribution: The key distribution problem, The Diffie-Hellman method, RSA and related methods, Linear cryptanalysis, Differential cryptanalysis, Meet-in-the-middle attacks, Symmetric &Asymmetric key together.

TEXT BOOKS

1. Stallings, ―Cryptography and Network Security- Principles and Practice‖, Pearson Ed., 2017.
2. Â NealKoblitz, ―A Course in Number Theory and Cryptography‖, Springer 2006
3. Jill Pipher, Jeffrey Hoffstein, Joseph H. Silverman, ―An Introduction to Mathematical Cryptography‖, Springer, 2008.

REFERENCES

1. Niven, Zuckerman and Montgomery, ―An Introduction to theory of numbers‖, Wiley 2006.
2. Kahate, ―Cryptography and Network Security‖, McGraw-Hill Higher Ed., 2009.

1. School offering the course	School of Computing
2. Course Code	**CSF353**
3. Course Title	**Foundation of Cyber Security**
4. Credits (L:T:P:C)	3:0:0:3
5. Contact Hours (L:T:P)	3:0:0
6. Prerequisites (if any)	NA
7. Course Basket	Discipline Elective

COURSE OUTLINE:
This course aims to give an outline of cyber security. The course will equip students with a vibrant view of the existing cyber security landscape considering not only technical measures and defenses, but also the other theme areas including legal, management, crime, risk, social and human factors.

COURSE OBJECTIVE:
To understand the crucial necessity of cyber security in computer systems, networks and enlighten numerous threat scenarios. To understand the well-known cyber-attack events, clarify the attack scenarios, and enlighten mitigation techniques. To understand the variance between Systems Cyber Security, Network Cyber Security, and cryptography, crypto-protocols etc. To analyses the cyber threats to critical structures.

COURSE OUTCOMES:
At the end of the course, the student will be able to:
CO1: Understand the cyber threat landscape, both in terms of recent developing issues and those issues which persist over time.
CO2: Outline the roles and effects of governments, commercial and other organizations, citizens and criminals in cyber security affairs.
CO3: Analyze the general values and policies that can be functional to systems to make them more vigorous to attack.
CO4: Choose key factors in cyber security from different corrective views including computer science, management, law, criminology

CURRICULUM CONTENT

UNIT-1: INTRODUCTION TO CYBER SECURITY (8L)
Overview of Cyber Security, Cyber Threats & Crime, Cyber Espionage, Internet Governance, Challenges and Constraints, necessity for a Comprehensive Cyber Security Policy, necessity for a Nodal Authority, necessity for an International convention on Cyberspace.

UNIT-2: SECURITY THREATS AND VULNERABILITIES (8L)
Overview, vulnerabilities in software, Intrusion, Physical Theft, Abuse of Privileges, Malware infection, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness.

UNIT -3: SECURITY PRACTICES & SECURITY SAFEGUARDS (7L)
Security Practices: Security Management, Security Policy, Risk Management, Information Classification Process, Security Procedures and Guidelines, Business Continuity and Disaster Recovery.

Security Safeguards: Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Anti-Malware software.

UNIT -4: INTRUSION DETECTION &SECURING WEB (7L)

Intrusion detection and Prevention Techniques, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges, Network based Intrusion detection & Prevention Systems.

UNIT -5: SECURITY LAWS, STANDARDS & FORENSIC (7L)

Security Laws & Standards: Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy,
Cyber Forensic: Digital Forensics Essentials: Principles of Digital Forensics and its challenges, Steps taken by computer forensic specialists, Benefits of Professional Forensic Methodology, Industry standards and reporting.

TEXT BOOKS

1. Micki Krause, Harold F. Tipton, ―Handbook of Information Security Management‖, Vol 1-3, CRC Press LLC, 2004.

REFERENCES

1. Bill Nelson, ―Computer Forensics and Investigations‖, Cengage Learning, India Edition, 2016.
2. Matt Bishop ―Computer Security Art and Science‖, Pearson/PHI, 2002.

1.School offering the course	School of Computing
2.Course Code	**CSF354**
3.Course Title	**Data Encryption and Network Security**
4.Credits (L:T:P:C)	2:0:1:3
5.Contact Hours (L:T:P)	2:0:2
6.Prerequisites (if any)	NA
7.Course Basket	DE

COURSE OUTLINE:
The course deals with the underlying principles of cryptography and network security. Starting from the classical encryption techniques to the more advanced tools of network security, the course imparts an immense coverage of the authentication and practices for securing network. The course deals with user/message authentication, IP security fundamentals. The course wraps up with the understanding of ACL, Firewalls and VPNs.

COURSE OBJECTIVE:
This course will cover the concept of security, types of attack experienced, encryption and authentication for deal with attacks, what is Network Perimeter Security, Access Control Lists and Virtual Private Networks.

COURSE OUTCOMES:
At the end of the course, the student will be able to:
CO1: Understand the significance of authentication process using digital signature.
CO2: Understand the significance of hash functions in data security.
CO3: Understand the concept of IP security and significance of Access control lists in network security.
CO4: Understand the concept of Communication Model, Network Perimeter Security Lists and Virtual Private Networks.

CURRICULUM CONTENT

UNIT 1: (6 L)
Symmetric & Asymmetric Key Cryptography: Algorithm types & Modes, Substitution and Transposition Ciphers
User Authentication Mechanism: Authentication basics, Passwords, Authentication tokens, Certificate based & Biometric authentication, Digital Signatures: Digital Signatures, authentication protocols, digital signature standards (DSS), proof of digital signature algorithm.

UNIT 2: (6 L)
Message Authentication and Hash Function: Approaches to Message Authentication, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions and MACS, MD5 message digest algorithm, secure hash algorithm (SHA).
Authentication Applications: Kerberos and X.509, directory authentication service, electronic mail security-pretty good privacy (PGP), S/MIME

UNIT 3: (5 L)
IP Security: Architecture, Authentication header, encapsulating security payloads, combining security associations, key management.
Network Perimeter Security Fundamentals: Introduction to Network Perimeter, Multiple layers of Network Security, Security by Router.

UNIT-4 **(5 L)**

Access Control Lists: Ingress and Egress Filtering, Types of Access Control Lists, ACL types: standard and extended, ACL commands.

Firewalls: Firewall Basics, Types of Firewalls, Network Address Translation Issues.

UNIT- 5: **(5 L)**

Virtual Private Networks: VPN Basics, Types of VPN, IPsec Tunneling, IPsec Protocols. VLAN: introduction to VLAN, VLAN Links, VLAN Tagging, VLAN Trunk Protocol (VTP).

TEXT BOOKS

1. Forouzan, B.A., ―Cryptography & Network Security‖, Tata McGraw-Hill Education, 2010.
2. Stallings, ―Cryptography and Network Security- Principles and Practice‖, Pearson Ed., 2017.

REFERENCES

1. Kahate, A., ―Cryptography and Network Security‖, McGraw-Hill Higher Ed., 2009.
2. Godbole, N., ―Information Systems Security: Security Management, Metrics, Frameworks and Best Practices‖ John Wiley & Sons India, 2009

1. School offering the course	School of Computing
2. Course Code	**CSF355**
3. Course Title	**Cyber Crime & Investigation**
4. Credits(L:T:P:C)	3:0:0:3
5. Contact Hours(L:T:P)	3:0:0
6. Prerequisites(ifany)	NA
7. Course Basket	DE

COURSE OUTLINE:

This course provides an impression of cybercrime and the investigation practices put in place to respond to them. The course will emphasis on the types and extent of present cybercrimes, how the justice system responds to these crimes, the various legal protections afforded to computer users, the regulation and policies that govern cybercrime detection and prosecution, and related machineries.

COURSE OBJECTIVE:

To describe the nature and area of cybercrime. To grow knowledge of key incidents of cybercrime and their subsequent influence. To study and debate national and global digital law implementation efforts. To categorize and assess the precise technology that enables cybercrime and digital law enforcement. To assess the influence of cybercrime on information professions.

COURSE OUTCOMES:

Having successfully completed this course, students will be able to reveal facts and understanding of:

CO1: Analyse the essential concepts of cybercrime and forensics.

CO2: Distinguish the object and causes for cybercrime, detection and handling.

CO3. Understand the extents affected by cybercrime and investigation.

CO4: Demonstrate tools used in cyber forensic and apply their knowledge for report writing

CURRICULUM CONTENT

UNIT-I: PRINCIPLES AND CONCEPTS OF CYBER CRIMINOLOGY (8 L)

Crime, Offence, Misdemeanour, Cyber Space, Cyber Crime, Cyber Criminology, Information Security, Penetration Testing, Incident Response, GRC, Conventional crimes vs. Cyber Crimes, White Collar Crimes, Economic Offences, Organized Crimes, Terrorism, Crime and Media and other contemporary forms of crimes.

UNIT-II: PSYCHOLOGY OF CYBER CRIMINALS (7 L)

Types of Cyber Criminals, Modus Operandi of Cyber Criminals, Profiling of Cyber Criminals, Tools and Techniques adopted by Cyber Criminals, Psychological theories relating to cyber criminals, Causes of Cyber Crimes, Criminological Theories and Cyber Crime, Routine Activity Theory, Social Learning Theory, Differential Association Theory, Differential Opportunity Theory, Media and Crime and latest theories and other related theories.

UNIT-III: DIGITAL INVESTIGATION (7L)

Digital Evidence and Computer Crime, History and Terminology of Computer Crime Investigation, Technology and Law, The Investigative Process, Investigative Reconstruction, Motive and Technology, Digital Evidence in the Courtroom.

UNIT-IV: COMPUTER FORENSIC & UNDERSTANDING INFORMATION (7L)

Computer Forensic Fundamentals: Applying Forensic Science to computers, Computer Forensic Services, Benefits of Professional Forensic Methodology, Steps taken by computer forensic specialists. **Methods of storing data:** number systems, character codes, record structures, file formats and file signatures, Word processing and graphic file formats, Structure and Analysis of Optical Media Disk Formats, Recognition of file formats and internal buffers, Extraction of forensic artefacts, understanding the dimensions of other latest storage devices, SSD Devices.

UNIT-V: TYPES OF COMPUTER FORENSICS TOOLS AND TECHNOLOGY (7L)

Tools and Types of Military Computer Forensics Technology, Tools and Types of Law Enforcement Computer Forensic Technology, Tools and Types of Business Computer Forensic Technology.

TEXT BOOKS

1. SunitBelapure and Nina Godbole. ―Cyber Security: Understanding Cyber Crime, Computer Forensic and Legal Perspectives‖, Wiley India Pvt Ltd, ISBN: 978-81-265-2179, publish date 2013.
2. Bil Nelson, Amelia Philips and Christopher Steuart, ―Guide to Computer Forensics and Investigation‖, 4th Edition, Cengage Learning 2015.

REFERENCES

1. Thomas J Mowbray, ―Cybersecurity Managing Systems, Conducting Testing, and investigating Intrusions‖, copyright 2014 by John Wiley & sons, ISBN: 978-1-118-84965, 2014.
2. James Graham, Ryan Olson, Rick Howard, ―Cyber Security Essentials‖, CRC press, 15 Dec 2010.

1. School offering the course	School of Computing
2. Course Code	**CSF445**
3. Course Title	**Mobile and Wireless Network Security**
4. Credits (L:T:P:C)	3:0:0:3
5. Contact Hours (L:T:P)	3:0:0
6. Prerequisites (if any)	NA
7. Course Basket	DE

COURSE OUTLINE:

This course will introduce students about Mobile and Wireless Networks, Vulnerabilities of Wired and Wireless Networks. It also includes overview of Fundamental Security Mechanisms, Hash functions, Electronic signatures and MAC, Cryptographic protocols. Topics would also include Wi-Fi Security Dedicated Architectures and Bluetooth Security.

COURSE OUTCOMES:

The purpose of this course is to provide In-depth knowledge about cellular design concepts and understanding of 3G Wireless network. It also provides an understanding of various security concerns and protocols in wireless networks (e.g., WiFi and mobile cellular networks) and mobile systems and applications.

COURSE OUTCOMES:

On successful completion of the course, students will be able to achieve the following:

CO1: Understand the security threats, vulnerabilities in wireless and mobiles systems and their related mechanisms

CO2 Understand the strategies for developing secure mobile applications.

CO3: Select mobile security penetration tools for evaluating the robustness of mobile applications.

CO4: Understand various models, design principles and solutions used in wireless network security to obtain authentication.

CURRICULUM CONTENT

Unit 1: Introduction to Mobile and Wireless Networks (8L)

Introduction, Cellular network basic concepts and Applications, First generation (1G) mobile, Second generation (2G) mobile, Third generation (3G) mobile, IEEE wireless networks, WLAN: IEEE 802.11, WPAN: IEEE 802.15, WMAN: IEEE 802.16, WMAN mobile: IEEE 802.20, MIH: IEEE 802.21,
WRAN: IEEE 802.22, Macro mobility, Micro mobility, NEMO and MANET networks

Unit 2: Vulnerabilities of Wired and Wireless Networks (7L)

Introduction, Security in the digital age, Threats and risks to telecommunications systems, Homogenity vs. heterogeneity, The Internet and security

Unit 3 Fundamental Security Mechanisms (7L)

Basics on security, Symmetric and asymmetric cryptography, Hash functions, Electronic signatures and MAC, Cryptographic protocols, Secure communication protocols and VPN Implementation, Secure Socket Layer (SSL) and Transport Layer Security (TLS), IPsec protocol suite, Comparison between

SSL and IPsec security protocols, IPsec VPN and SSL VPN, Authentication, Access control, Firewalls, Intrusion detection

Unit-4 Wi-Fi Security Dedicated Architectures **(8L)**

Introduction, Hot spot architecture: captive portals, Captive portal overview, Security analysis, Wireless intrusion detection systems: architecture, events, example; Wireless honeypots: design, requirements.

Wi-Fi Security: Introduction, Attacks on wireless networks, Passive attacks, Active attacks, TCP attacks, Trojan attack, Dictionary attacks, Security in the IEEE 802.11 standard, IEEE 802.11 security mechanisms, WEP (Wired Equivalent Privacy), WEP shortcomings, Attacks, Security in 802.1x, Authentication in wireless networks, RADIUS, EAP authentication procedures, PKI, Level 3 VPN, IPsec

Unit- 5: **Bluetooth Security** **(7L)**

Introduction, Organization of Bluetooth nodes in the network, Bluetooth technical specification, Radio physical layer, Baseband, Link controller, Bluetooth device addressing, HCI layer, L2CAP layer, Bluetooth security, Bluetooth encoding, Attacks.

TEXT BOOKS

1. Jonathan Katz and Yehuda Lindell, Introduction to Modern Cryptography, Chapman & Hall/CRC Cryptography and Network Security Series, 2nd edition 2014.
2. Frank Adelstein, Sandeep K.S. Gupta, Golden G. Richard III, and Loren Schwiebert, Fundamentals of Mobile and Pervasive Computing, 2005.

REFERENCES

1. Levente Buttyán and Jean-Pierre Hubaux, Security and Cooperation in Wireless Networks, 2008.
2. James Kempf, Wireless Internet Security: Architectures and Protocols, 2008.
3. PatrickTraynor, Patrick McDaniel, and Thomas La Porta, Security for Telecommunications Networks, 2008.

1. School offering the course	School of Computing
2. Course Code	**CSF446**
3. Course Title	**Ethical Hacking and Digital Forensics**
4. Credits (L:T:P:C)	2:0:1:3
5. Contact Hours (L:T:P)	2:0:2
6. Prerequisites (if any)	NA
7. Course Basket	DE

COURSE OUTLINE:

This course will introduce students about Hacking windows, Network hacking, Password hacking, TCP / IP – Checksums, Dos attacks – SYN attacks, Smurf attacks, UDP flooding, DDOS Models. Firewalls, Packet filter firewalls, Packet Inspection firewalls, Application Proxy Firewalls. Batch File Programming, Fundamentals of Computer Fraud, Strategic Planning Process, Architecture strategies for computer fraud prevention, Penetrating testing process, Key Fraud Indicator selection process customized taxonomies, Computer Forensics, Accounting Forensics, Journal risk and control matrix, Misuse detection and Novelty detection

COURSE OBJECTIVE:

This course provides an introduction the concepts of Ethical Hacking and provides an understanding of Computer forensics fundamentals. This course will provide the opportunity to learn about different tool and techniques in Ethical Hacking and will analyse various computer forensics technologies and methods for data recovery.

COURSE OUTCOMES:

At the end of the course, the student will be able to:

CO1: Identify and analyse the stages an ethical hacker requires in order to compromise a target system.

CO2: Understand the concepts of computer forensics fundamentals and types of computer forensics.

CO3: Evaluate security techniques used to protect system and user data.

CO4: Illustrate the methods for data recovery, evidence collection and data seizure.

CURRICULUM CONTENT

UNIT 1: (6 L)

Hacking windows – Network hacking – Web hacking – Password hacking. A study on various attacks – Input validation attacks – SQL injection attacks – Buffer overflow attacks - Privacy attacks.

UNIT 2: (6 L)

TCP / IP – Checksums – IP Spoofing port scanning, DNS Spoofing. Dos attacks – SYN attacks, Smurf attacks, UDP flooding, DDOS – Models. Firewalls – Packet filter firewalls, Packet Inspection firewalls – Application Proxy Firewalls. Batch File Programming.

UNIT 3: (5 L)

Fundamentals of Computer Fraud – Threat concepts – Framework for predicting inside attacks – Managing the threat – Strategic Planning Process.

UNIT-4 **(5 L)**

Architecture strategies for computer fraud prevention – Protection of Web sites – Intrusion detection system – NIDS, HIDS – Penetrating testing process – Web Services – Reducing transaction risks.

UNIT- 5: **(5 L)**

Key Fraud Indicator selection process customized taxonomies – Key fraud signature selection process – Accounting Forensics – Computer Forensics – Journaling and it requirements – Standardized logging criteria – Journal risk and control matrix – Neural networks – Misuse detection and Novelty detection.

TEXT BOOKS

1. Stuart McClure, Joel Scambray and Goerge Kurtz, ―Hacking Exposed 7: Network Security Secrets & Solutions‖, Tata McGraw Hill Publishers, 2010.
2. Bensmith, and Brian Komer, ―Microsoft Windows Security Resource Kit‖, Prentice Hall of India, 2010.

REFERENCES

1. Kenneth C.Brancik, ―Insider Computer Fraud‖ Auerbach Publications Taylor & Francis Group, 2008.
2. Ankit Fadia, ― Ethical Hacking‖ 2nd Edition Macmillan India Ltd, 2006 MTCF -202 Database S
3. Stuart McClure, Joel Scambray and Goerge Kurtz, ―Hacking Exposed Network Security Secrets & Solutions‖, 5th Edition, Tata McGraw Hill Publishers, 2010.

Computer Vision and Biometrics

S. No.	Course Code	Course Title (Proposed)	Credits: L T P C
1.	CSF344	Machine Learning	2 0 2 3
2.	CSF356	Digital Image Processing	2 0 2 3
3.	CSF357	Satellite Image Processing	2 0 2 3
4	CSF358	Computer Vision	2 0 2 3
5.	CSF447	Information Retrieval	2 0 2 3
6.	CSF448	Biometrics Security	2 0 2 3

1.	School offering the course	School of Computing
2.	**Course Code**	**CSF356**
3.	**Course Title**	**Digital Image Processing**
4.	**Credits (L:T:P:C)**	2:0:1:3
5.	**Contact Hours (L:T:P)**	2:0:2
6.	**Prerequisites (if any)**	NA
7.	**Course Basket**	DE

COURSE SUMMARY

This course will introduce students to the basis of digital image processing, various types of image models, and conversion from one model to another. They can learn about spatial and frequency domain models for image processing and will be able to implement various image enhancement techniques like filtering, object extraction. They will also be able to understand the classification of objects through feature extraction.

COURSE OBJECTIVES

The participants will learn the basic concepts of digital image processing, working with images using spatial and frequency domain, implementing various image enhancement techniques like filtering to an image using these domains. Moreover, the classification of content presents in an image through objects detection and feature extraction from the given input image will be clear.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the basis of image processing for the enhancement of color images.
CO2: Understand the spatial domain and frequency domain approaches of digital image processing.
CO3: Implement various techniques associated with image filtering i.e. smoothing and sharpening.
CO4: Implement the concepts of classification through object detection followed by feature extraction.

CURRICULUM CONTENT

Unit-1: Introduction (5L)

Motivation and Perspective, Applications, Components of Image Processing System, Element of Visual Perception, a Simple Image Model, Sampling and Quantization. Image Enhancement in Spatial Domain;
Basic Gray Level Functions – Piecewise-Linear Transformation Functions: Contrast Stretching; Histogram Specification; Histogram Equalization; Local Enhancement; Enhancement using Arithmetic/Logic Operations – Image Subtraction, Image Averaging; Basics of Spatial Filtering; Smoothing - Mean filter, Ordered Statistic Filter; Sharpening – The Laplacian.

Unit-2: Image Enhancement in Frequency Domain (4L)

Fourier Transform and the Frequency Domain, Basis of Filtering in Frequency Domain, Filters – Lowpass, High-pass; Correspondence Between Filtering in Spatial and Frequency Domain; Smoothing Frequency Domain Filters – Gaussian Low pass Filters; Sharpening Frequency Domain Filters – Gaussian high pass filters; Homomorphism Filtering.

Unit-3: Segmentation (5L)

Region Extraction, Pixel-Based Approach, Multi-level Threshold, Local Threshold, Region-based
Approach, Edge and Line Detection: Edge Detection, Edge Operators, Pattern Fitting Approach, Edge Linking, and Edge Following, Edge Elements Extraction by Threshold, Edge Detector Performance, Line Detection, Corner Detection, Image Registration.

Unit-4: Color Image Processing **(6L)**

Color Fundamentals, Color Models, and Converting Colors to different models, Color Transformation, Smoothing and Sharpening, Color Segmentation. Morphological Image Processing: Introduction, Logic Operations involving Binary Images, Dilation and Erosion, Opening and Closing, Morphological Algorithms – Boundary Extraction, Region Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening

Unit-5: Feature Extraction **(6L)**

Representation, Topological Attributes, Geometric Attributes Description, Boundary-based Description, Region-based Description, and Relationship. Object Recognition, Deterministic Methods, Clustering, Statistical Classification, Syntactic Recognition, Tree Search, Graph Matching

TEXTBOOKS:

1. Digital Image Processing Rafael C. Gonzalez, Richard E. Woods Prentice Hall, 2007 (3rd Edition)

REFERENCES:

1. Robert J. Schalkoff, Digital Image Processing, and Computer Vision, John Wiley and Sons, NY, 1989. 1st Edition.
2. Anil K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Upper Saddle River, NJ

1.	School offering the course	School of Computing
2.	Course Code	**CSF357**
3.	Course Title	**Satellite Image Processing**
4.	Credits (L: T:P:C)	2:0:1:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

This course will introduce students to the basis of digital image processing, various types of image models, and conversion from one model to another. They can learn about spatial and frequency domain models for image processing and will be able to implement various image enhancement techniques like filtering, object extraction. They will also be able to understand the classification of objects through feature extraction.

COURSE OBJECTIVES

The objective of the course is to de about the procedure of satellite data acquisition and analysis. Moreover, interpretation and classification of content present in a satellite image through objects detection and feature extraction from the given input image.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Select the type of remote sensing techniques/data for the required purpose.
CO2: Identify the earth's surface features from satellite images.
CO3: Analyse the energy interactions in the atmosphere and earth surface features.
CO4: Get familiar with various image enhancement and image processing techniques.

CURRICULUM CONTENT

Unit 1: Fundamentals (5L)

Remote Sensing Components Electro-Magnetic Spectrum; Radiometric quantities; Atmospheric window; Spectral reflectance of vegetation, soil and, water–atmospheric influence on spectral response patterns; Satellite systems and data-acquisition-storage-orbits-Data formats-Data products-Image processing system-factors to be considered-Image display systems-Image sampling and quantization Basic relationship between pixels.

Unit 2: Sensor and Data Model (5L)

Classification of remote sensors – selection of sensor parameters - resolution concept - Spectral, Radiometric and temporal resolution – Image formation – Histogram - spatial statistics – Image registration and ortho- rectification - Geometric and radiometric correction. Quality of images in optical systems – imaging mode – photographic camera – optomechanical scanners – push broom and whiskbroom cameras – Panchromatic, multispectral, hyperspectral scanners – geometric characteristics of scanner, imagery - Landsat, SPOT, IRS, World View.

Unit 3: Image Enhancements (5 L)

Spectral signatures – Image characteristics, feature space sscattergram point, local and regional operation – spatial feature and multi-image manipulation techniques - principle component analysis - Optimal Rotation Transformation – Scale-space transforms, wavelet transform. Multi-image fusion. Sources of errors in received data – referencing scheme – data product output medium – GeoTIFF, and HDF formats.

Unit-4 : **Information Extraction** **(5L)**

Data products and their characteristics – Elements of visual interpretation – Digital image processing – Pre-processing – Image rectification – Image enhancement techniques – Image classification – Supervised and unsupervised classification algorithms for multispectral and hyperspectral images – Accuracy assessment. parametric Classification -Decision tree – other Non - parametric classifiers - subpixel and super-pixel classification.

Unit- 5: Image Analysis and Change Detection **(6L)**

Pattern recognition - boundary detection and representation - textural and contextual analysis - decision concepts: Fuzzy sets - evidential reasoning -Hyperspectrall image analysis – Accuracy assessment. Expert system - Artificial Neural Network – Case studies General Steps required while performing Change Detection, Land-Use/Land Cover Classification System, Remote Sensing System Consideration, and Change Detection Algorithm.

TEXTBOOKS :

1. Lilliesand and T.M. and Kiefer, R.W., "Remote Sensing and Image Interpretation ", John Wiley and Sons, 1994
2. Mather M. Paul “Computer Processing of Remotely-Sensed Images: An Introduction”, 3rd Edition, 2005.

REFERENCES :

1. Charles Elachi and Jakob J. van Zyl Introduction To The Physics and Techniques of Remote Sensing, Wiley Series in Remote Sensing and Image Processing, 2006.
2. George Joseph, Fundamentals of Remote Sensing, Second Edition, Universities Press (India) Pvt Ltd, Hyderabad, 2005, ISBN: 8173715351, 9788173715358
3. Sabins, F.F.Jr, Remote Sensing Principles and Image interpretation, W.H.Freeman& Co,1978

1.	School offering the course	School of Computing
2.	Course Code	**CSF358**
3.	Course Title	**Computer Vision**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

Computer vision is the science and technology of machines that can see. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems. Various research areas include: Applications in Display Technology, Computer Vision for Navigation, Metrology, High Level Video Analysis, and Human-Computer Interfaces.

COURSE OBJECTIVES

This course is designed to provide knowledge about computer vision algorithms, methods, and concepts; which will enable the students to implement computer vision systems with an emphasis on applications and problem-solving.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: To recognize and identify specific faces among others.

CO2: Learn how to install OpenCV and explore basic image processing concepts.

CO3: To develop techniques to separate foreground and background in images, create stunning panoramas, calibrate camera and automatically detect common objects like faces or people in images.

CO4: To build a 3D representation of a scene using stereoscopic images.

CURRICULUM CONTENT

Unit 1: (4 L)

Introduction of Image Formation, Geometric Camera Models, Light and Shading, Human Color perception, Linear filters, Local image features, texture. Binary Image Analysis and Segmentation: Properties, digital geometry, Segmentation. Machine learning for machine vision: Learning and inference in vision, modeling complex data densities, regression models, Classification models.

Unit 2: (6 L)

Image segmentation by clustering: Basic Clustering methods, watershed algorithm, segmentation using K-means, Mean Shift: Finding Local Modes in Data, Clustering, and Segmentation with Mean Shift, Segmentation, Clustering and Graph, Hough Transformation. Motion segmentation: Optical flow and motion, flow models, motion segmentation with layers; Model Selection: Cross-Validation.

Unit 3 (8 L)

Tracking: Tracking by detection, Tracking translation by matching, Affine transformation; The Kalman filter, Forward-backward Smoothing; Data association; Particle filtering Classification Strategies: Mahalanobis distance, Class-Conditional histograms, Naïve Bayes, Nearest Neighbours, Linear Support vector machine, Kernel Machines, Boosting and AdaBoost Object detection in Images: Sliding window methods: Face detection, detecting Humans, detecting boundaries; detecting deformable Objects.

Unit-4 **(8 L)**

Image processing for feature detection and Image synthesis, edge detection, corner detection, line and curve detection, SIFT operator, Image-based modeling and rendering, Mosaics, snakes Stereo: shape from shading, photometric stereo, texture, Occluding contour detection, motion analysis: Motion detection and optical flow structure from motion; Object recognition: Hough transforms and other simple object recognition.

TEXT BOOKS:

1. David A. Forsyth and Jean Ponce. Computer Vision: A Modern Approach. Second Edition Pearson 2015.
2. Robert Haralick and Linda Shapiro. Computer and Robot Vision. Vol-I/II, Addison Wesley, 1993.

REFERENCES:

1. Milan Sonka, Vaclav Hlavac, and Roger Boyle. Image Processing, Analysis, and Machine Vision. Fourth Edition. CENGAGE Learning.

1.	School offering the course	School of Computing
2.	Course Code	**CSF447**
3.	Course Title	**Information Retrieval**
4.	Credits (L: T:P:C)	3:0:0:3
5.	Contact Hours (L: T:P)	3:0:0
6.	Prerequisites (if any)	
7.	Course Basket	DE

COURSE SUMMARY

Information retrieval is the process through which a computer system can respond to a user's query for text-based information on a specific topic. IR was one of the first and remains one of the most important problems in the domain of natural language processing (NLP). Web search is the application of information retrieval techniques to the largest corpus of text anywhere, the web and it is the area in which most people interact with IR systems most frequently

COURSE OBJECTIVES

The aim is to give students an understanding of the fundamental techniques for hypermedia architectures, design and usability, document management and retrieval, metadata management, and searching the web. In this course, we will cover basic and advanced techniques for building text-based information systems, including the Efficient text indexing, Boolean and vector-space retrieval models, Evaluation and interface issues, IR techniques for the web, including crawling, link-based algorithms, and metadata usage, Document clustering and classification, Traditional and machine learning-based ranking approaches.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Demonstrate the basic elements of representing and retrieving documents.
CO2: Understand the technologies for linking, describing, and searching the web.
CO3: Design an information retrieval system for web hypermedia
CO4: Apply and create e relationship between IR, hypermedia, and semantic models.

CURRICULUM CONTENT

Unit 1: Introduction to Information Retrieval (7 L)

The nature of unstructured and semi-structured text. Inverted index and Boolean queries. Text Indexing, Storage and Compression; Text encoding: tokenization, stemming, stop words, phrases, index optimization. Index compression: lexicon compression and postings list compression. Gap encoding, gamma codes, Zapf's Law. Index construction. Postings size estimation, merge sort, dynamic indexing, positional indexes, n-gram indexes, real-world issues.

Unit 2: Retrieval Models (8 L)

Boolean, vector space, TFIDF, Okapi, probabilistic, language modeling, latent semantic indexing. Vector space scoring. The cosine measure. Efficiency considerations. Document length normalization. Relevance feedback and query expansion.

Unit 3 Performance Evaluation (7 L)

Evaluating search engines. User happiness, precision, recall, F-measure. Creating test collections: kappa measure, interjudge agreement. Text Categorization and Filtering: Introduction to text classification. Naive Bayes models. Spam filtering. Vector space classification using hyperplanes; centroids; k Nearest Neighbours. Support vector machine classifiers. Kernel functions. Boosting.

Unit-4: Text Clustering (7 L)

Clustering versus classification. Partitioning methods. K-means clustering. A mixture of Gaussians model. Hierarchical agglomerative clustering. Clustering terms using documents.

Unit-5: Web Information Retrieval **(7 L)**

Hypertext, web crawling, search engines, ranking, link analysis, PageRank, HITS. Retrieving Structured Documents, XML retrieval, semantic web

Text Books

1. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to Information Retrieval, Cambridge university press, 2017

Reference Books

1. David Lowe and Wendy hall, Hypermedia and the Web: An Engineering Approach, John Wiley, 1999, ISBN: 0-417-98312-8
2. R.K. Belew, Finding out about--A cognitive perspective on search engine technology and the www, Cambridge University Press, 2001

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF448**
3.	Course Title	**Biometric Security**
4.	Credits (L: T:P:C)	3:0:0:3
5.	Contact Hours (L: T:P)	3:0:0
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

Biometrics is the science of identifying or authenticating an individual's identity based on behavioral or physiological characteristics. Government Ids, secure electronic banking, retail sales, and health and social services all have benefited from the use of biometric technology and will continue to do so as biometric research advances. This course introduces students to the basic principles and methods used for biometric identification. The objective is to provide students with the scientific foundations needed to design, implement, and evaluate large-scale biometric identification systems.

COURSE OBJECTIVES

The aim is to give students an understanding of biometric systems based on a number of biometric traits such as the face, fingerprint, iris, and hand shape. In this course, we will cover basic and advanced techniques for biometrics applications using MATLAB, biometric system modalities such as face recognition, fingerprint recognition, iris recognition, hand shape recognition, Biometric system design, and performance evaluation, multi-modal biometric systems, and privacy and ethical issues.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Basic information on the fundamental physical and organic science and designing standards of biometric frameworks.

CO2: Understand biometric frameworks and be able to examine and design for essential biometric framework applications.

CO3: Understand various Biometric security issues.

CO4: Describe Cryptography security and Fuzzy models.

CURRICULUM CONTENT

Unit 1: (9 L)

Introduction- Authentication systems, Development of biometric authentication. Basic terms, biometric data, biometric characteristics, biometric features, biometric templates and references. Expected properties of biometric identifiers. Basics in biometric errors estimation. Enrolment, verification and identification. How Authentication Technologies work, Benefits of biometrics over traditional authentication systems, How Biometrics work. Applications of Biometrics.

Unit 2: (9 L)

Fingerprints and Hand Geometry: Technical description, Characteristics, Competing technologies, Strengths–Weaknesses, Deployment. Face and Voice Recognition: Technical description, Characteristics, Strengths-Weaknesses, Deployment.

Unit 3 (9 L)

Biometric System Security: Secure transfer of biometric data. Secure storage, use of smart cards, principles of match-off-card and match-on-card techniques. Biometrics in the cloud. Points of attack. Privacy models. Spoofing: Static and dynamic liveness features. Liveness detection in biometrics. Selected liveness detection techniques, frequency analysis for paper printouts detection.

Unit-4 **(9 L)**

Protection: Overview of principles from cryptography to secure fuzzy data. Template protection strategies: feature protection, key-binding, key-generating, hybrids. Overview of fuzzy vaults, fuzzy commitment, fuzzy extractors and revocable bio tokens. Bio cryptographic infrastructures for secure template management.

TEXT BOOKS :

1. John D. Woodward, Jr. Nicholas M. Orlans Peter T. Higgins, "Biometrics", dream tech, 2003
2. Samir Nanavathi, Michel Thieme, Raj Nanavathi,"Biometrics -Identity verification in a network", Wiley Eastern, 2002

REFERENCES :

1. John Chirillo and Scott Blaul," Implementing Biometric Security", Wiley Eastern Publications, 2005

Cloud Computing and Blockchain

S. No.	Course Code	Course Title	Credits: L T P C
1.	CSF349	Cloud Computing	2 0 2 3
2.	CSF354	Data Encryption and Network Security	2 0 2 3
3.	CSF361	Introduction to Blockchain Technologies	2 0 2 3
4	CSF362	Design & Development of Blockchain Technologies	2 0 2 3
5.	CSF363	Blockchain Ecosystems & Governance	2 0 2 3
6.	CSF364	Container Technologies	2 0 2 3

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF349**
3.	Course Title	**Cloud Computing**
4.	Credits (L: T:P:C)	2: 0: 2 :3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

This course is designed to teach students the basic concepts and terminology of cloud computing. After establishing the definition of cloud computing, this course describes the various service delivery models of cloud computing architecture, and the ways in which clouds can be deployed as public, private, hybrid and community clouds. Students also learn about the security challenges that cloud deployments experience, and how these are addressed.

COURSE OBJECTIVES

This course is designed to provide knowledge about basic concepts of Cloud computing. Public cloud and its service and deployment models, private cloud and its need and challenges, Multi-cloud and business cloud, security threats in the cloud.

COURSE OUTCOMES

Course Outcomes (COs): After the completion of the course, students will be able to:

CO1: Elaborate cloud computing its service and deployment models.

CO2: Formulate the importance of virtualization, multi-tenancy in the cloud environment.

CO3: Define and examine different cloud computing services.

CO4: Categorize the different security threats and challenges faced by cloud provider, and Demonstrate the different types of business cloud and its uses.

CURRICULUM CONTENT

UNIT 1 **(6L)**

Overview of cloud computing and Distributed Computing: What is a cloud, Definition of cloud, Definition of cloud, characteristics of cloud, Traditional vs. Cloud Computing, Importance of Cloud Computing, Cloud service models (IaaS, PaaS & SaaS). Cloud deployment models (Public, Private, Hybrid and Community Cloud), Benefits and Challenges of Cloud Computing. Introduction, Examples of distributed computing, Concurrent Programming, Characteristics & Properties of Distributed Systems, client-server model, centralized vs distributed computing, Resource Sharing and the Web Challenges, security issues.

UNIT 2 **(7L)**

Private Cloud: Concept of Hypervisor, Basics of virtualization, Virtualization technologies, Server virtualization, VM migration techniques, Role of virtualization in Cloud Computing. Business cases for the need of Cloud computing environment, Concept of Private Cloud, Characteristics of Private Cloud, Private Cloud deployment models, Private Cloud Vendors, Virtual Private Cloud. Multitenancy, Types of tenancy, Application programming interfaces (API), Billing and metering of services.

UNIT 3 **(7L)**

Public Cloud: Concept of Public Cloud, Importance of Public Cloud, when to opt for Public Cloud, Public Cloud Service Models, and Public Cloud players. Infrastructure as a Service Offerings, IaaS Vendors, PaaS offerings, PaaS vendors, Software as a Service. Implementing public cloud AWS, Introduction, Service Offered, Creation of EC2 instance, Microsoft Azure: Introduction, Service Offered, Creation of DB instance. Implementing Security in public Cloud, Comparison of Public Cloud Vendors (AWS, Microsoft, Google, IBM, Salesforce).

UNIT 4 **(6L)**

Multi-Cloud: Concept of multi-cloud management, Challenges in managing heterogeneous clouds, benefits of multi-cloud management systems. Case study on Multi-Cloud Management System (Right Scale Cloud Management System). Business Clouds: Cloud Computing in Business, Various Biz Clouds focused on industry domains (Retail, Banking and Financial sector, Life Sciences, Social networking, Telecom, Education).

UNIT 5 **(7L)**

Cloud Security: Cloud security reference model, Principal security dangers/risks to cloud computing, Internal security breaches, Data corruption, Malicious Insiders, Data Loss or Leakage, Account or Service Hijacking, Unknown Risk Profile, Steps to reduce cloud security breaches, Identity management: Detection and forensics, Identity management: Detection and Identity management, Benefits of identity, Encryption techniques, Encryption & Encrypting data, Attacks on VM, Abuse and Nefarious Use of Cloud Computing.

TEXTBOOK(S)

1. R. Buyya, C. Vecchiola, S. T. Selvi, Matering Cloud Computing, Ed. Third reprint, 2013.
2. B. Sosinsky, Cloud computing Bible, Ed. Reprint Willy India Pvt. Ltd, 2014.
3. Carlin, Sean, and Kevin Curran. "Cloud computing security." Pervasive and Ubiquitous Technology Innovations for Ambient Intelligence Environments. IGI Global, 2013. 12-17.

REFERENCES :

1. M. Miller, Cloud Computing, Pearson education in South Asia, Ed. 9th 2014.
2. Buyya, Rajkumar, James Broberg, and Andrzej M. Goscinski, eds. Cloud computing: Principles and paradigms. John Wiley & Sons, 2010.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF361**
3.	Course Title	**Introduction to Blockchain Technologies**
4.	Credits (L: T:P:C)	2:1:0:3
5.	Contact Hours (L: T:P)	2:1:0
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The students will learn the basic components of blockchain, concept of distributed system, consensus mechanism and its important and basic of cryptocurrency.

COURSE OBJECTIVES

This course objective is to explain basic components of a blockchain (types, mechanics: transaction, block, block header, chain and terminology) its operations (processes, verification, validation, and consensus model) underlying algorithms, and essentials of trust to understand how blockchain systems (mainly Bitcoin and Ethereum) work.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1: Recognize foundational concepts of blockchain and learn about the decentralized peer-to-peer network.
CO2: Understand the formal definition of distributed consensus and apply these concepts on the blockchain.
CO3: Assess Blockchain applications in a structured manner.
CO4: Understand the meaning and properties of crypto economics: cryptography and economics.

CURRICULUM CONTENT

Unit 1: (6L)

Distributed Systems: Blockchain architecture, Basic components (blocks, nodes, etc.), Distinction between public and private blockchains, benefits and drawbacks, Fundamental traits and characteristics, Distributed Database, CAP theorem, the Byzantine Generals Problem and Fault Tolerance.

Unit 2: (6L)

Cryptography in Blockchain: Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing Complete. Blockchain Network, Mining Mechanism

Unit-3: (6L)

Consensus: Distributed Consensus, Merkle Patricia Tree, Gas Limit, Cryptography: Hash function, Digital Signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof, Transactions and Fee.

Unit 4: (6L)

Blockchain Design Principle: Consensus, Security, and Operating Protocols, Blockchain Design Principle, Public and Private DLTs, Alternative Consensus Mechanisms to Bitcoin's Proof-of-Work, Proof-of-Stake, Proof-of-Burn, Voting-Based Consensus Algorithms, and Federated Consensus, Sybil Attack, Energy Utilization.

Unit-5 (6L)

Crypto economics: Property of crypto economics: cryptography and economics, Integration of cryptography and pseudo-anonymity in public blockchains, cryptoeconomics with respect to distributed systems fundamentals (liveness, safety, data availability).

TEXT BOOKS:

1. Kube N. Daniel Drescher: Blockchain basics: a non-technical introduction in 25 steps, 2018.

REFERENCES:

1. Warburg B, Wagner B, Serres T. Basics of Blockchain: A Guide for Building Literacy in the Economics, Technology, and Business of. Animal Venturs LLC; 2019..

Tutorial: Naive Blockchain construction, Memory Hard algorithm - Hashcash implementation, Direct Acyclic Graph.

1.	School offering the course	School of Computing
2.	Course Code	**CSF362**
3.	Course Title	**Design & Development of Blockchain Technologies**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The students will learn the basics of Ethereum platform, smart contracts and its purpose and programming in solid language.

COURSE OBJECTIVES

This course aims to educate students, Ethereum, basics of smart contracts, decentralized apps, decentralized anonymous organizations (DAOs), and solidity as a programming language.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the Ethereum platform and Dapps and DAOs
CO2: Understand smart contracts and its designing structure.
CO3: Design Smart contracts using solidity programming language.
CO4: Define decentralized appas and its applications.

CURRICULUM CONTENT

UNIT 1 **(4L)**

Introduction to Ethereum: concepts of Smart Contracts, Dapps, And DAOs, Ethereum Virtual Machine (EVM).

UNIT 2 **(4L)**

Ethereum Technology: Overview, Architectural Overview Ethereum Block chain Platform, Current and Potential Uses of Ethereum.

UNIT 3 **(8L)**

Introduction to Programming Smart Contracts: A Simple Smart Contract, Structure of a Contract, Types, Units and Globally Available Variables, Input Parameters and Output Parameters, Control Structures, Function Calls, Order of Evaluation of Expressions, Assignment, Scoping and Declarations, Error handling: Assert, Require, Revert and Exceptions.

UNIT 4 **(8L)**

Solidity Programming: Basics of Solidity, Layout of a Solidity Source File & Creating Contracts, General Value Types (Int, Real, String, Bytes, Arrays, Mapping, Enum, address), Visibility and Getters, Function Modifiers, Constant State Variables, Functions, Inheritance, Abstract Contracts, Interfaces, Libraries.

UNIT 5 **(6L)**

Introduction to Decentralized Apps (Dapps): Decentralized Application Architecture, connecting to the Block chain and Smart Contract, Decentralized Apps – Coding Details, Voting Contract, Coding Style Guide, Design Patterns, Coding Style Guide, Code Layout, Naming Conventions, Common Design Patterns, Withdrawal from Contracts, State Machine.

TEXT BOOKS :

2. Antonopoulos AM, Wood G. Mastering ethereum: building smart contracts and dapps. O'reilly Media; 2018 Nov 13.
3. Bistarelli S, Mazzante G, Micheletti M, Mostarda L, Sestili D, Tiezzi F. Ethereum smart contracts: Analysis and statistics of their source code and opcodes. Internet of Things. 2020 Sep 1;11:100198.

REFERENCES :

1. Troxell C. Writing Smart Smart Contracts. Access on 4th March 2022.
2. Solorio K, Kanna R, Hoover DH. Hands-on Smart Contract Development with Solidity and Ethereum: From Fundamentals to Deployment. O'Reilly Media, Incorporated; 2019.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF363**
3.	Course Title	**Blockchain Ecosystems & Governance**
4.	Credits (L: T:P:C)	2:1:0:3
5.	Contact Hours (L: T:P)	2:1:0
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The students will learn the comparison between traditional and blockchain system. Some cases of blockchain implementation with some advantages and disadvantages, future of blockchain, etc.

COURSE OBJECTIVES

This course enables the students to understand the broader blockchain ecosystem, other blockchain platforms, application use cases, and challenges such as privacy and scalability

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:

CO1- Understand permissioned blockchain, and concept of hyperledge and Blockchain as a service.

CO2- Define various use cases on realtime applications using blockchain and concept of cryptocurrency.

CO3- Understand the concepts of regulations and anonilty and legal aspects attached with technology.

CO4- Define and understand the future of blockchain globally and Indian scenario.

CURRICULUM CONTENT

UNIT 1 **(6L)**

Enterprise Blockchain: Real-World Applications: Permissioned Blockchains: The Linux Foundation's Hyperledger and Microsoft Azure's Blockchain as a Service, JP Morgan's Quorum, Ripple, and Tendermint.

UNIT 2 **(6L)**

Blockchain use Cases: Challenges and solutions Applications of blockchain, Business and industry use cases: cybersecurity, the integrity of information, E-Governance, Climate Change, Biodiversity, Energy, Internet of Things, Medical Record Management System, Sustainability and other contract enforcement mechanisms etc.

UNIT 3 **(6L)**

Scaling Blockchain (Cryptocurrency): Bitcoin as a payment method, comparison with traditional forms, Cryptocurrency: History, Distributed Ledger, Bitcoin protocols - Mining strategy and rewards, Vulnerability, Attacks, Namecoin, Vertical scaling (e.g. block size increases, Segregated Witness and the Lightning Network), Horizontal Scaling (e.g. sidechains, sharding).

.

UNIT-4 **(6L)**

Regulation and Anonymity: Anti-Money Laundering (AML) and Know Your Customer (KYC) Regulations, Anonymity goals, and government techniques for deanonymization of entities on blockchain, stakeholders, Roots of Bitcoin, Legal Aspects - Cryptocurrency Exchange, Black Market and Global Economy.

UNIT 5 **(6L)**

Blockchain Future: Global Status on Blockchain, Historic stance of the Indian Government, Current Scenario, Myths vs reality of blockchain technology, Understanding and working knowledge of the emerging blockchain technology**,** Do-main Name Service and future of Blockchain: venture capitalism, ICOs, and crowdfunding.

TEXT BOOKS:

1. Jun, M. Blockchain government - a next form of infrastructure for the twenty-first century. J. open innov. 4, 7 (2018). https://doi.org/10.1186/s40852-018-0086-3
2. Nijalingappa, Pradeep, and Mangesh Manikrao Ghonge, eds. Blockchain Technologies and Applications for Digital Governance. IGI Global, 2021.

REFERENCES:

1. Savirimuthu J. Blockchain and the law: The rule of code., 2019

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF364**
3.	Course Title	**Container Technologies**
4.	Credits (L: T:P:C)	2: 0: 2 :3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

This course is designed to teach students the basic concepts and terminology of cloud computing. After establishing the definition of cloud computing. This course describes the basics of container technologies used in cloud computing, dockers, concept of Orchestration and kurbernetes

COURSE OBJECTIVES

This course provides the opportunity to learn concepts and design Containerization and build an Orchestration of containers. It also provides an ability to promote the cost effective light weight virtualization using container orchestration management tools and techniques.

COURSE OUTCOMES

Course Outcomes (CO): After the completion of the course, students will be able to:

CO1. Elaborate the container technology

CO2. Formulate and Design containers using Docker.

CO3. Categorize and demonstrate the concept of containerization using Docker files and Compose

CO4. Categorize and design an Orchestration of nodes.

CURRICULUM CONTENT

UNIT 1 **(6L)**

Introduction Container Technology: Containerization, History of Containers, Namespaces and C-groups, Containers vs Virtual Machines, Types of Containers. Docker: Overview, Installing Docker on Linux, Installation, Hub, Images, Containers, Features of Docker, Components of Docker.

UNIT 2 **(7L)**

Creating Containerized Services: Working with Containers, Architecture, Container & Hosts, Configuring, Containers & Shells, File, Building Files, Public Repositories, Managing Ports, Private Registries, Building a Web Server Docker File.

UNIT 3 **(7L)**

Managing Containers: Instruction Commands, Container Linking, Storage, Networking, Setting Node.js, Setting MongoDB, Setting NGINX, Toolbox, Setting ASP.Net, Docker Cloud, Logging, Docker – Compose, Docker - Continuous Integration.

UNIT 4 **(7L)**

Orchestration in Docker: Create and run multi-container applications using Docker Compose and manage clusters of Docker nodes using Docker Swarm. Topics: Docker Compose, Docker Swarm, Docker Service, Placement Rolling Update and Rollback Docker Stack, deploy a Multi-container Application using Compose, Running Docker in Swarm mode, deploying a Service in Swarm Scale, Services, Service Placement, Rolling Updates and Rollbacks Docker Stack.

UNIT 5 **(6L)**

Introduction to Kubernets: Understanding Kubernetes architecture, Introduction to Kubernetes objects, using basic Kubernetes objects, Using the kubectl command, Leveraging Kubernetes.

TEXT BOOKS:

1. Antonopoulos, Nick, and Lee Gillam. Cloud computing. London: Springer, 2010.
2. Comer, Douglas E. The Cloud Computing Book: The Future of Computing Explained. Chapman and Hall/CRC, 2021.
3. Raj, Pethuru, Jeeva S. Chelladhurai, and Vinod Singh. Learning Docker. Packt Publishing Ltd, 2015.
4. Luksa, Marko. Kubernetes in action. Simon and Schuster, 2017.

REFERENCES :

1. Foster, Ian, and Dennis B. Gannon. Cloud computing for science and engineering. MIT Press, 2017.
2. Chaudhary, Sanjay, Gaurav Somani, and Rajkumar Buyya, eds. Research advances in cloud computing. Springer Singapore, 2017.
3. Turnbull, James. The Docker Book: Containerization is the new virtualization. James Turnbull, 2014.
4. Sayfan, Gigi. Hands-On Microservices with Kubernetes: Build, deploy, and manage scalable microservices on Kubernetes. Packt Publishing Ltd, 2019.
5. Protechgurus, Dockers containers ultimate beginners guide, Independently published , August 18, 2018

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

Full Stack and DevOps

S. No.	Course Code	Course Title	Credits: L T P C
1.	CSF349	Cloud Computing	2 0 2 3
2.	CSF364	Container Technologies	2 0 2 3
3.	CSF371	Front-End Engineering	2 0 2 3
4	CSF372	Advance Topics in Front-End Engineering	2 0 2 3
5.	CSF373	Server Side Engineering	2 0 2 3
6.	CSF374	DevOps	2 0 2 3

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF371**
3.	Course Title	**Front End Engineering**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L:T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The students will learn the basic terminologies related to web applications. They will be able to design and develop web applications using JAVA technologies.

COURSE OBJECTIVES

To become familiar with components of front-end web application development: User interfaces, Event and State handling, Languages/tools such as HTML, CSS, JavaScript.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the concept of technology used to design a simple web page.
CO2: Demonstrate the use script and events handling in web page.
CO3: Demonstrate the process to connect with server.
CO4: Design an application to store data on remote location and access it.

CURRICULUM CONTENT

UNIT 1: Introduction to HTML (4 L)

HTML Basics, Elements, Attributes, Styles, Forms, Form Elements, Input Element Types, Input Attributes, File Paths, Script tag, HTML & XHTML.

UNIT 2: Introduction to CSS (4 L)

CSS Introduction, Syntax, Selectors, Styling, Pseudo class, Pseudo Elements, CSS Tables, CSS Box Models, CSS Opacity, CSS Navigation Bar, Dropdowns.

UNIT 3: Introduction to JavaScript (4 L)

JavaScript Statements, Keywords, Functions, JavaScript Programs, Operators, Functions, Function Parameters, Function Return Types, Data Types, Primitive Types.

UNIT 4: NodeJS and Application Desigm (10 L)

Introduction to Node JS: Introduction to Node JS, Advantages of Node JS, What is Node JS, Node.js Process Model, Traditional Web Server Model, Node JS Modules: Functions, Buffer, Module, Modules Types, Core Modules, Local Modules, Modules Exports, Node Package Manager: What is NPM, Installing Packages Locally, Installing package globally, Adding dependency in package json, Updating packages, Creating Web Server: Creating Web Server, Sending Requests, Handling http requests, File System: reading, writing, updating files, and the concept of chunks, buffers, and uploading files synchronously and asynchronously.

UNIT 5: Introduction to MongoDB (10 L)

Overview of MongoDB, Design Goals for MongoDB Server and Database, MongoDB tools, Understanding the following: Collection, Documents and Key/ Values, etc., Schema Design and Data Modelling Goal: Manage Data Model in MongoDB. Skills, Understand Data Modelling Schemas, Design Data Model relationships and tree structures, Apply Data Modelling in various real-time contexts, CRUD Operations.

TEXTBOOK(S)

1. Mark Sapp, Front-end Web Developer (Careers in Technology Series): JavaScript, HTML5 and CSS3, Addison Wesley, 2018.
2. Bruno Joseph D'Mello, Mithun Satheesh, Jason Krol, Web Development with MongoDB and Node, Pact Publishing, 3rd Edition, 2017.

REFERENCES:

1. Julie Meloni, Jennifer Krynin, Sams Teach Yourself HTML, CSS and JavaScript All in One, Pearson, 3rd Edition, 2015.
2. Jennifer Robbins, Learning Web Design: A Begginer's Guide to HTML, CSS< JavaScript and Web Graphics, O'Reilly, 5th Edition, 2018.

1.	School offering the course	School of Computing
2.	Course Code	**CSF372**
3.	Course Title	**Advanced Concepts in Front End Engineering**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	NA
7.	Course Basket	DE

COURSE SUMMARY

The students will learn the advanced concepts related to designing of web applications.

COURSE OBJECTIVES

Provide opportunity to design a full fledge one-spage web-application.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the concepts advance JAVA script.
CO2: Understand and design an interface using class and object of advance JAVA script.
CO3: Demonstrate the use of style sheet in one-page application.
CO4: Design and develop a complete MVC application.

CURRICULUM CONTENT

UNIT 1: Advance JAVA Script (6 L)

Revising basic concept of JavaScript, Function Hoisting, Function within Function, Function Expressions Passing function as arguments, Mouse and Keyboard Events , Propagation of Event ,Closures, const and let, Let in for loops, Arrow Functions , Bindings in Arrow Function.

UNIT 2: Classes and Constructors (6 L)

This Keyword, This in Strict Mode, Function to create Objects , Object constructor, Adding Behaviour to Objects, Objects, Class, Properties and Methods, Class Expressions and Hoisting , Exports and Imports declaration.

UNIT 3: Introduction to React (4 L)

What is React, What are components, SPA vs MPAs, React vs others, Tools and Installation of tools, The terminal Create-react-app, Folder structure, Installing react developer tools, Understanding JSX, JSX Restrictions, Creating a Functional Component.

UNIT 4: Styling React components and Elements (12 L)

Working with Components & Re-Using Them, Outputting ,Dynamic Content, Working with Props, Understanding the "children" Prop, Understanding & Using State, Props & State, Handling Events with Methods, Events Listening, Manipulating the State, Function Components Naming, Using the use State() Hook for State Manipulation, Stateless vs Stateful Components, Passing Method References Between Components, Adding Two Way Binding, Adding Styling with Stylesheets, Working with Inline Styles, Working with list and conditions, Outlining the Problem Set, Setting Styles Dynamically, Setting Class Names Dynamically, Adding and Using Radium, Using Radium for Media Queries, Introducing Styled Components, More on Styled Components, Styled Components & Dynamic Styles, Working with CSS Modules, CSS Modules & Media Queries, React Hooks.

TEXTBOOK (S):

1. Julie Meloni, Jennifer Krynin, Sams Teach Yourself HTML, CSS and JavaScript All in One, Pearson, 3rd Edition, 2015.
2. Robun Wieruch, The Road to REACT, CreateSpace Independent Publishing Platform, 2017.

REFERENCES :

1. Alex Banks, Eve Porcello, Learning React: Modern Patterns for Developing React Apps, O'Reilly, 2nd Edition, 2020.
2. Zac Gordon, React Explained, OS Training, 2019.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1.	School offering the course	School of Computing
2.	Course Code	**CSF373**
3.	Course Title	**Server-Side Engineering**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	None
7.	Course Basket	Discipline Elective

COURSE SUMMARY

The students will gain familiarity with what server-side programming is and what it can do

COURSE OBJECTIVES

Provide Opportunity to student to learn the concepts of MVC application and design a full fledge CRUD application.

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand and design an application using Maven.
CO2: Design an MVC application using Spring framework.
CO3: Design and develop a Restfull service.
CO4: Design a CRUD based application.

CURRICULUM CONTENT

UNIT 1: Maven (6 L)

What is Maven, Why command line, Dependency Resolution, Configurations, Installation Approach - Archetype, RAD,Setup Commands, Download from GIT, Life cycles, Phases and Goals, Profiles, Parent-Child Module, Dependency Plugins, Local Maven Repository Vs Project Centralise Repository.

UNIT 2: Spring Framework Overview (8 L)

Introduction to Spring, Installation, first application, Spring Container and Dependency Injection: Spring Container types, Working of Spring container, Dependency Injection by Constructor, Injecting string-based values. Bean Implementation: Introduction and Scope, creating source file, Implement Collections, Implement Java Based Configuration. **Aspect Oriented Programming**: Spring AOP, Implement Aspect Oriented Programming.

UNIT 3: Spring Data Access and Transaction Management (4 L)

Spring JDBC, JDBC Template, Spring Transaction. Develop Web Application using Spring: Spring Web MVC Overview, Advanced MVC Features, Development of Spring Web Application

UNIT 4: Spring ORM (12 L)

Hibernate + JPA, Introductions, DataSource Configurations, Object Relational, Mapping, About Maven Dependencies and Configs, Entity classes and Session Factory, CRUD Operations – APIs, Hibernate Configurations, Session Factory, Sessions, Mapping XML, Entities, Annotation Based, First/Second Level Caching , Transient, Persistent and Detached Objects

TEXTBOOK(S) :

1. Balaji Varanasi, Introducing Maven: A Build Tool for Today's JAVA Developers, Apress, 2nd Edition, 2019.
2. Craig Walls, Spring in Action, Manning Publications, 5th Edition, 2019.

REFERENCES :

1. Marten Deinum, Daniel Rubio, Josh Lang, Gary Mak, Spring 5 Recipes: A Problem-Soluiton Approach, Appress, 4th Edition, 2017.

1.	School offering the course	School of Computing
2.	Course Code	**CSF374**
3.	Course Title	**DevOps**
4.	Credits (L: T:P:C)	2:0:2:3
5.	Contact Hours (L: T:P)	2:0:2
6.	Prerequisites (if any)	None
7.	Course Basket	Discipline Elective

COURSE SUMMARY

The students will learn the aspects of the principles of continuous development and deployment, software development operations, continuous integration, automation of configuration management and learn the various tools like Git, Docker, Jenkins, Ansible etc.

COURSE OBJECTIVES

Provide opportunity to students to learn concepts of Devops and tools used at different stages of Software Automations

COURSE OUTCOMES

On successful completion of the course, students will be able to achieve the following:
CO1: Understand the concept of DevOps.
CO2: Understand the concept of retrieval and operation on project file at remote location.
CO3: Understand about integration of complete project using Jenkins.
CO4: Learn to configure node resources using Ansible.

CURRICULUM CONTENT

UNIT 1: Introduction to DevOps (10 L)

Principle, DevOps Engineer Skills in the market, Delivery Pipeline, Market trend of DevOps, Technical Challenges, Tools use in DevOps, CALMR Model. DevOps and Other Frameworks: Agile Framework, Lean Framework, Waterfall Model, Scrum / Kanban Framework, DevOps Roles and Considerations: DevOps Roles, DevOps Responsibilities In An Organization DevOps Improvements, DevOps Practices: RACI Model, RCA Process, DevOps And Automation, Continuous Integration, Continuous Testing, Continuous Delivery / Deployment, Continuous Monitoring, Continuous Feedback.

UNIT 2: Version Control using GIT (10 L)

Git – A CLI, Essentials of GIT in industry, How to setup GIT, Installing Git, First-Time Git Setup, Getting a Git Repository, Working with various commands in GIT, Recording Changes to the Repository How to check the Status of Your Files, How to track New Files, Staging our modified files, Ignoring Files from GIT, Viewing Your Unstaged and Staged Changes How to commit Your Changes, Skipping the Staging Area and commit, Removing Files from GIT, Viewing the Commit History, Limiting Log Output, Using a GUI to Visualize History, Undoing Things, Changing Your Last Commit, Unstaging a Staged File.

UNIT 3: Continuous Integration with Jenkins (7 L)

An Overview Of JENKINS, Getting Started with JENKINS and Installation of JENKINS, Plugins and Its Uses, Setting Up Your Build Jobs, Using Metrics to Improve Quality, Nodes and Master-Slave Configuration, Performing Automated Deployment and Continuous Delivery, Pipeline Execution Of CI CD Jobs, JENKINS Administration Activities.

UNIT 4: Infrastructure Management using Ansible (7 L)

Introduce Ansible, Describe the terminology and architecture of Ansible, Deploy Ansible Install Ansible and run ad hoc commands, Write Ansible plays and execute a playbook. Manage variables and inclusions Describe variable scope and precedence, manage variables and facts in a play, and manage inclusions. Implement task control Manage task control, handlers, and tags in Ansible playbook.

TEXTBOOK(S):

1. Emily Freeman, DevOps for dummies, Wiley, 2019.
2. Scott Chacon, Ben Straub, Pro Git, Apress, 2nd Edition, 2014.
3. John Ferguson Smart, Jenkins: The Definitive Guide, O'Reilly, 2011.
4. Michael Heap, Ansible: From Beginner to Pro, Apress, 2016

REFERENCES:

1. Michael Hutterman, DevOps for Developers, Apress, 2012, doi: https://doi.org/10.1007/978-1-4302-4570-4 (Accessed: March 2022).
2. Jesse Liberty, Jon Galloway, Git for Programmers: Master Git for effective implementation of version control for your programming projects, Packt publishing, 2021.
3. Jonathan McAllister, Mastering Jenkins, Packt Publishing, 1st Edition, 2015.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1. Department offering the course	Computer Science and Engineering
2. Course Code	CSF101
3. Course Title	Programming for problem solving
4. Credits (L:T:P:C)	3:0:2:4
5. Contact Hours (L:T:P)	3:0:2
6. Prerequisites (if any)	None
7. Course Basket	Engineering Sciences

COURSE OUTLINE:

This course contains the fundamental concepts about the computer hardware and intends to provide to students about the knowledge of C language

COURSE OBJECTIVE:

The objective of the course is to make the students to understand the key hardware components in a modern computer system and as to how the software is mapped to the hardware. The student shall also be able to learn make the computer programs using C language by exploring the various features of C.

COURSE OUTCOMES:

At the end of the course, the student will be able to:

CO1. Develop simple algorithms for arithmetic and logical problems.

CO2. Implement conditional branching, iteration and recursion.

CO3. Describe a problem into functions and synthesize a complete program using divide and conquer approach.

CO4. Implement arrays, pointers and structures to formulate algorithms and programs.

CURRICULUM CONTENT

UNIT 1: Introduction to Computer, Programming & algorithms (8 L)

Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.)

Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples, From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code

UNIT 2: Arithmetic Expression, and Conditional statements, Loops, Expression: (7 L)

Arithmetic, Logical, Relational expressions and precedence.

Loops & Branching: Writing and evaluation of conditionals and consequent branching, Iteration and loops.

UNIT 3: Arrays & Functions (7 L)

Arrays: Arrays (1-D, 2-D), Character arrays and Strings.

Functions: functions (including using built in libraries), Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference, Storage class.

Searching & Sorting: Searching, Basic Sorting Algorithms (Bubble sort)

UNIT 4: Recursion and Structure (8 L)

Recursion: Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc.

Structure: Structures, Defining structures and Array of Structures.

UNIT 5: Pointers & File handling **(7 L)**

Pointers: Idea of pointers, Defining pointers, Use of Pointers in self-referential structures
File handling: different modes of opening a file in C, reading, writing from files.

TEXT BOOKS

1. Byron Gottfried, "Schaum's Outline of Programming with C", 2nd edition 2006 McGraw-Hill.

2. E. Balaguruswamy, "Programming in ANSI C", 8th Edition 2019, McGraw-Hill Education India.

REFERENCES

1. Brian W. Kernighan and Dennis M. Ritchie, "The C Programming Language", 2nd edition 1988, Prentice Hall of India.

LIST OF EXPERIMENTS:

S.NO.	EXPERIMENT NAME
1	Familiarization with programming environment.
2	Programming for Simple computational problems using arithmetic expressions.
3	Programming for Problems involving if-then-else structures.
4	Programming for Iterative problems e.g., sum of series.
5	Programming for 1-D Array manipulation.
6	Programming for Matrix problems, String operations.
7	Programming for Simple functions
8	Programming for Recursive functions.
9	Programming for Pointers and structures.
10	Programming for File operations

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2022-26

1. School offering the course	School of Computing
2. Course Code	**CSF102**
3. Course Title	**Data Structures**
4. Credits (L:T:P:C)	3:0:1:4
5. Contact Hours (L:T:P)	3:0:2
6. Prerequisites (if any)	CSF101 (Programming for problem solving)
7. Course Basket	Engineering sciences

Course Summary: The course is a foundation level course and requires the knowledge of the C programming language. The course outlines the detailed architecture and implementation of basic data structures such as Stacks, Queues, Linked Lists, Trees, and Graphs. It also covers the time and space complexity analysis of different searching and sorting techniques. Some of the searching methods include Linear Search, Binary Search, and sorting mechanism includes Bubble sort, Insertion sort, Selection sort, Quick sort, Merge sort, and Heap Sort. The course also incorporates different hashing techniques, designing hash functions, hash table implementation, and collision resolution technique.

Course Objectives: The main objective of this course is to introduce the concept of data structure, how to choose a particular data structure, and how the choice of a data structure impacts the performance of programs. Other objectives include:
CO1: How to select the appropriate data structure model specific to some application.
CO2: Solve problems using data structures like Stacks, Queues, Linked Lists, Trees, Graphs, and writing programs for these solutions using C code.
CO3: Introduce the concept of algorithm writing, analyzing algorithms, converting pseudocode to appropriate C code, and showing how one solution is better than others by analyzing their computational complexities.

Course Outcomes: On successful completion of the course, students will be able to:
CO1: Develop an ability to read, write, and analyze the time and space complexity of any algorithms.
CO2: Describe the properties, behaviour, and implementation of basic data structures like Stacks, Queues, Linked List, Trees, and Graphs.
CO3: Convert pseudocode to its appropriate C code implementation.
CO4: Compare different searching and sorting techniques in terms of their memory usage and time consumption.
CO5: Design and implement different hash functions, analyze the collision effect, and hash table implementations.

Curriculum Content

Unit I: Introduction to Algorithms & Data Structure (8 L)
Introduction: Data types, Abstraction, Abstract Data Type (ADT), Concept of data structure, Types of data structures, Operations on Data Structures, Introduction to Algorithms, Writing Pseudocodes, Algorithm analysis, Complexity of algorithms and Time space trade-off, Searching: Linear and Binary Search Techniques and their complexity analysis.

Unit II: Arrays, Stacks, and Queues (7 L)
Arrays: Introduction to Array, Applications of Array, Operations on Arrays: Traverse, Insert, Delete etc. Stacks: Introduction to Stacks, Array representation of Stack, Operations on Stack: Push, Pop, etc. Applications of Stacks: Infix and Postfix Conversion, Evaluations of Infix and Postfix expressions. Queue: Introduction to Queue, Array representation and implementation of queues, Operations of Queue, Applications of Queue, Types of Queue: Circular Queue, Priority Queue, Double ended Queue. Operations on each type of Queue and their Applications.

Unit III: Linked Lists and Trees (8 L)
Linked Lists: Introduction to Dynamic Memory Allocation, Representation and Implementation of Single, Double, and Circular Linked Lists, Operations on Linked List: Insert, Delete, Traverse etc. Applications of Linked List, Linked List representation of Stack and Queue. Trees: Basic Tree terminologies, Types of Trees: Binary Tree, Binary Search Tree (BST), AVL Tree, B-Tree, and Heap. Representation and Implementations of different types of trees, Tree Traversal algorithms, Operation on trees: Insert, Delete, etc., Applications of Tress.

Unit IV: Graphs **(7 L)**

Graphs: Introduction to Graph and their Terminologies, Types of Graph, Representations of Graph, Graph traversal algorithms, Topological Sorting, Minimum Spanning Tree, Shortest Path Algorithms: Single Source Shortest Path like Bellman-Ford, Dijkstra and All Pair Shortest Path like Floyd-Warshall.

Unit V: Sorting & Hashing **(9 L)**

Sorting Algorithms and their Analysis: Selection Sort, Bubble sort, Insertion sort, Quick sort, Merge sort, Heap Sort. Performance Analysis and Comparison of all sorting techniques. Hashing: Hash Functions and its type, Hash Table construction, Collision Resolution, Universal Addressing, Open Hashing.

Text Books

1. Aaron M. Tenenbaum, Yedidyah, Langsam, Moshe J. Augenstein, Data Structures using C Pearson. 1st Edition. 2019
2. Schaum's outline series, Data structures with C, McGraw Hill Education; 1st edition (July 2017)

Reference Books

1. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publication,2nd Edition. 2008. 2. Robert Kruse, Data Structures and Program Design in C PHI.2nd Edition.2006.
3. Willam J. Collins, Data Structure and the Standard Template library –2003, T.M.H.1st Edition.
4. Kyle Loudon, Mastering Algorithms with C, O'Reily Publication, 1st Edition, 1999

List of Experiments:

1. Program in C for the implementation of Array for various operations.
2. Program in C for the creation of Stack for its various operation implementation.
3. Program in C for the creation of Queue for its various operation implementation.
4. Program in C for the creation of Link list for its various operation implementation.
5. Program in C for the creation of Circular Link list for its various operation implementation.
6. Program in C for the creation of Doubly Link list for its various operation implementation.
7. Program in C for the creation of Binary Search Tree for its various operation implementation.
8. Program in C for the Implementation of sorting Algorithms.
9. Program in C for the Implementation of basic Graph Algorithms.