DIT UNIVERSITY
Dehradun

Detailed Course Structure & Syllabus

of

B.Tech. – Computer Science & Engineering

With Specializations in

AI, Machine Learning and Robotics
Data Science and Analytics
Internet of Things and Edge Computing
Cybersecurity and Privacy
Computer Vision and Biometrics
Cloud Computing and Blockchain*
Full Stack and DevOps*

(Fully Flexible Choice Based Credit System)

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Introduction

The Ministry of Human Resource Development (MHRD), Govt. of India, has initiated development of a 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 courses in 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.

8. 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.

9. 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.

10. 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.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

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 by a 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 will be 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: 2020-24

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

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.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

DIT University CSE 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**

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

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
	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
	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
MEF201	Mechanical Engineering Materials	3	0	2	4
PEF204	Fluid Mechanics	3	0	2	4
EEF141	Electrical Engineering Material	3	0	2	4
ECF142	Fundamental of Semiconductor Electronics	3	0	2	4
CEF101	Civil Engineering Materials	3	1	0	4

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

	Skill Enhancement (min 8 credits to be taken)				
	Name of Courses	L	T	P	C
CSFXXX	Technical Training 1	0	0	4	2
CSFXXX	Technical Training 2	0	0	4	2
CSFXXX	Value Added Training 1	0	0	4	2
CSFXXX	Value Added Training 2	0	0	4	2
SWAYXXX	MOOCS Courses (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
LAF381	Positive Psychology & Living	3	0	0	3
LAF481	Application of 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
LAF482	Intellectual Property Rights	3	0	0	3
LAF382	Engineering Economics	3	0	0	3
	Free Electives (min 9 credits to be taken)				
	Name of Courses	L	T	P	C
ECF481	Analogue Electronics (ECE)	2	0	2	3
ECF482	Cellular Communication Network (ECE)	2	0	2	3
ECF381	Microcontroller (ECE)	2	0	2	3
ECF382	Bio Medical Instrumentation (ECE)	2	0	2	3
ECF483	Digital Image processing (ECE)	2	0	2	3
CSF381	Software Project Management	3	0	0	3
CSF345	Introduction to Data Science	3	0	0	3
CSF482	Introduction to Cyber security	3	0	0	3
MEF381	Composites materials	3	0	0	3
MEF481	Total Quality Management	3	0	0	3
MEF482	Renewable Energy Sources	3	0	0	3
PEF381	Carbon Capture and Sequestration	3	0	0	3
PEF491	Polymer Technology	3	0	0	3
PEF492	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
	Project (12 credits)				
UCF439	Capstone Project	0	0	12	12

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Course Baskets: B.Tech. CSE FFCBCS DC and DE Baskets

	Discipline Core (48 credits)					
	Contact Hrs Credits					
	Name of Courses	**Pre-requisite Courses**	**L**	**T**	**P**	**C**
CSF201	Computer Organization and Architecture	CSF101	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	CSF202	3	0	2	4
CSF206	Advanced Java Programming	CSF203	3	0	2	4
CSF301	Software Engineering	CSF102, CSF203	3	0	2	4
CSF302	Design and Analysis of Algorithms	CSf102	3	0	2	4
CSF303	Computer Networks	CSF206	3	0	2	4
CSF304	Artificial Intelligence	CSF302	3	0	2	4
CSF305	Theory of Computation	CSF202	3	1	0	4
CSF401	Compiler Design	CSF305	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**
CSF344	Machine Learning		2	0	2	3
CSF342	Fuzzy Logic and Neural Network		2	0	2	3
CSF343	Evolutionary Computing		2	0	2	3
CSF442	Robotics		2	0	2	3
CSF441	Deep Learning		2	0	2	3
CSF341	R Programming		2	0	2	3
	Data Science and Analytics					
CSF345	Introduction to Data Science		2	0	2	3
CSF346	Data Mining and Data Warehousing		2	1	0	3
CSF344	Machine Learning		2	0	2	3
CSF443	Big Data Analytics		2	0	2	3
CSF441	Deep Learning		2	0	2	3
CSF341	R Programming		2	0	2	3
	Internet of Things and Edge Computing					
CSF351	Advanced Computer Networks		3	0	0	3
CSF347	Wireless and Mobile Systems		3	0	0	3
CSF349	Cloud Computing		2	0	2	3
CSF445	Mobile & Wireless Network Security		2	0	2	3
CSF444	Internet of Things		2	0	2	3
CSF348	Mobile Application Programming using Android		2	0	2	3
	Cyber security and Privacy					
CSF352	Number Theory and Cryptology		2	0	2	3

CSF353	Foundation of Cyber Security	2	1	0	3
CSF354	Data Encryption & Network Security	2	0	2	3
CSF355	Cyber Crime & Investigation	3	0	0	3
CSF446	Ethical Hacking & Digital Forensics	2	0	2	3
CSF445	Mobile & Wireless Network Security	3	0	0	3
Computer Vision and Biometrics					
CSF356	Digital Image Processing	2	0	2	3
CSF358	Computer Vision	2	0	2	3
CSF447	Information Retrieval	3	0	0	3
CSF341	Machine Learning	2	0	2	3
CSF357	Satellite Image Processing	2	0	2	3
CSF448	Biometrics Security	3	0	0	3

Flowchart of pre-requisites for the DC courses
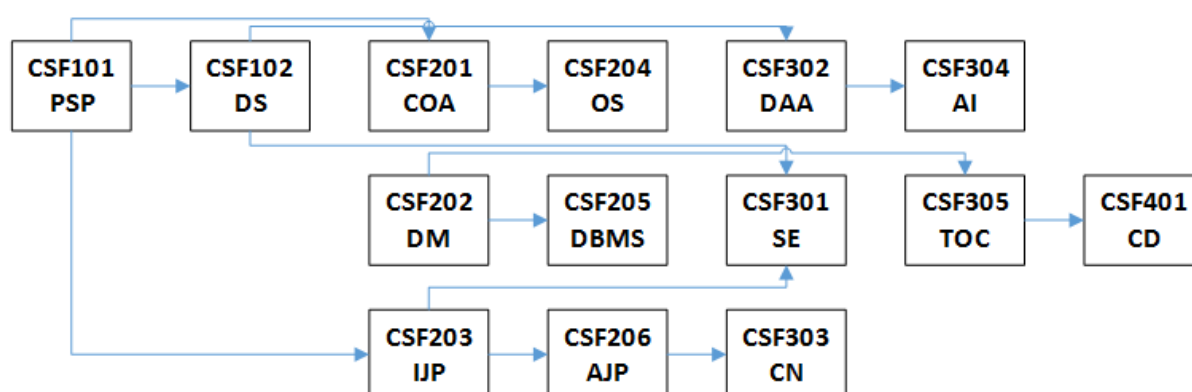
Abbreviations

1	PSP	Problem Solving and Programming
2	DS	Data Structures
3	COA	Computer Organisation & 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

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

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 SAP and in LMS 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

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

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)	3:0:1:4
5.	Contact Hours (L:T:P)	3:0:2
6.	Prerequisites (if any)	
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. To formulate simple algorithms for arithmetic and logical problems.

CO2. To implement conditional branching, iteration and recursion.

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

CO4. To use arrays, pointers and structures to formulate algorithms and programs.

CO5. To apply programming to solve matrix addition and multiplication problems and searching and sorting problems

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.

Searching & Sorting: Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

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

TEACHING AND LEARNING STRATEGY
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

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
11	Programming for solving Numerical methods problems

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	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. The other objective may 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 achieve the following:

CO1: The student will develop an ability to read, write, and analyze the time and space complexity of any algorithms.

CO2: Able to describe the properties, behaviour, and implementation of basic data structures like Stacks, Queues, Linked List, Trees, and Graphs.

CO3: Able to convert pseudocode to its appropriate C code implementation.

CO4: Able to compare different searching and sorting techniques in terms of their memory usage and time consumption.

CO5: Able to design and implement different hash functions, analyze the collision effect, and hash table implementations.

CURRICULUM CONTENT

Unit 1: 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 2: 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 3: 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.

Unit4: 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- 5: 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.

TEXTBOOK(S)
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

TEACHING AND LEARNING STRATEGY
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

List of Experiments

S.NO.	EXPERIMENT NAME
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.

COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
2.	Course Code	CSF201
3.	Course Title	Computer Organization and Architecture
4.	Credits (L:T:P:C)	3:1:0:4
5.	Contact Hours (L:T:P)	3:1:0
6.	Prerequisites (if any)	Programming for problem solving
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 -

CO1: This will help the students to be familiarized with the hardware components and concepts related to the control design

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

CO3: Students will learn about various I/O devices and the I/O interface.

CO4: The student will be able to learn the hardware components and concepts related to the memory organization.

CO5: Students will be able to get the theoretical concept of parallel processing and different types of multiprocessor's interconnection structures.

CURRICULUM CONTENT

Unit-1 Fundamental of Computer Design (8 L)

Basic Structure of Computers: Computer Types; Functional Units; Bus structure; Performance- Processor Clock, Basic Performance Equation, Clock rate; Historical Perspective; Machine Instructions and Programs: Numbers, Arithmetic Operations and Characters; Memory Location and Addresses; Memory Operations; Instructions and Instruction Sequencing

Unit-2 Instruction set, Assembly language and input/output Organization (9 L)

Machine Instructions and Programs: Addressing Mode; Assembly Language; Basic input and Output Operations; Stacks and Queues; Subroutines; Encoding of Machine Instructions; Accessing I/O Devices; Interrupts- Interrupt Hardware; Enabling and Disabling Interrupts; Handling Multiple Devices; Controlling Device Requests; Exceptions; Direct Memory Access; Standard I/O Interfaces-PCI Bus, SCSI Bus, USB.

Unit-3 Pipelining and Parallel Processing   (8 L)

Introduction to Pipelining; pipeline hazards; Implementation of pipeline; Instruction level parallelism concepts and challenges: Basic compiler techniques for exposing ILP; Reducing branch costs with prediction; Overcoming data hazards with dynamic scheduling; hardware-based speculation; Exploiting ILP using multiple issue and static scheduling; Introduction to multicore architecture

Unit-4 The Memory System   (6 L)

Basic Concepts: Semiconductor RAM Memories, Read only memories, speed, size, and cost, cache memories- mapping functions, replacement algorithms; cache performance; cache optimization; Virtual memory; Protection: Virtual memory and virtual machines.

Unit-5 Arithmetic for Computers   (8 L)

Addition and subtraction of signed numbers, design of fast adders, multiplication of positive numbers, signed operand multiplication, fast multiplication, integer division, floating-point numbers and operations.

TEXT BOOKS:

1.   John P. Hayes. Computer Architecture and Organization, 3rd Edition, McGraw Hill, 2012
2.   M. Morris Mano. Computer System Architecture 3rd Ed, Pearson Education-2017
3.   Carl Hamacher, Zvonko Vranesic, Safwatzaky. Computer Organization, 5th Edition, TMH 2011

REFERENCE BOOKS:

1.   John L. Hennessey and David A. Patterson: Computer Architecture, A Quantitative Approach, 4th Edition, Morgan Kaufmann;2006
2.   Kai Hwang: Advanced Computer Architecture Parallelism, Scalability, Programmability, 2nd Edition, Tata Mc Graw Hill, 2010

TEACHING AND LEARNING STRATEGY

All materials (pdf/ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	
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: An ability to perform operations on discrete structures such as sets, functions, relations, and sequences.

CO2: An ability to construct proofs using direct proof, proof by contradiction, proof by cases, and mathematical induction.

CO3: An ability to demonstrate the ability to solve problems using counting techniques and combinatorics in the context of discrete probability.

CO4: An ability to solve problems involving recurrence relations and generating functions.

CO5: An ability to prove computational theorem

CO6: Understand and implement various types of searching, sorting and hashing.

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

TEXTBOOK(S)

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

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>UNDERGRADUATE COURSE DESCRIPTION DOCUMENT</u>

1.	Department offering the course	Computer Science and Engineering
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)	Programming for problem solving
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, and strings, loops,

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

CO4: Understand and implement object-oriented paradigm using abstraction, encapsulation, inheritance, and polymorphism.

CO5: Understand and implement the objects and classes, exception handling, abstract classes, interfaces, Text I/O and binary I/O.

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 (12)

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, The 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, The Array List Class.

UNIT5: Exception handling, Abstract Classes and Interfaces, Binary I/O. Generics (6 L)

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.

Introduction, How Is Text I/O Handled in Java?, Text I/O vs. Binary I/O, Binary I/O Classes

Motivations and Benefits, Defining Generic Classes and Interfaces, Generic Methods.

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

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

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 text I/O and binary I/O.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Introduction to Java Programming
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, networking, java database programming. Students will also be introduced to technologies like Java beans, Servlets and JSP.

COURSE OBJECTIVES
The objectives of this course are to learn advanced java programming techniques and technologies required to build applications at enterprise level with good user interface.

COURSE OUTCOMES
On successful completion of the course, students will be able to achieve the following:
CO1: Understand and implement JavaFX technologies.
CO2: Understand and implement java collections framework.
CO3: Understand and implement multithreading in Java.
CO4: Understand and implement networking and java database programming.
CO5: Understand and implement enterprise level technologies like beans, servlets and jsp.

CURRICULUM CONTENT

UNIT I: user Interface (7 L)
Java FX vs Swing and AWT, The Basic Structure of a Java FX Program, Panes, UI Controls, and Shapes, The Color Class, The Font Class, The Image and Image View Classes, Layout Panes, Shapes.
Events and Event Sources, Registering Handlers and Handling Events, Inner Classes, Anonymous Inner Class Handlers, Simplifying Event Handling Using Lambda Expressions, Mouse Events, Key Events, Listeners for Observable Objects.
Labeled and Label, Button, Check Box, Radio Button, Text Field, Text Area, Combo Box, List View, Scrollbar, Slider.

UNIT 2:Java Collections Framework *(7 L)*
Collections, Iterators, Lists, the Comparator Interface, Static Methods for Lists and Collections, Queues and Priority Queues, Binary Search Trees, Array Lists, Linked Lists, Queues, Maps.

UNIT 3: Multithreading *(8 L)*
Thread Concepts, Creating Tasks and Threads, The Thread Class, Thread Pools, Thread Synchronization, Synchronization Using Locks, Cooperation among Threads, Case Study: Producer/Consumer, Blocking Queues, Semaphores, Avoiding Deadlocks, Thread States, Synchronized Collections.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

UNIT4: Networking and Java database programming (6 L)

Client/Server Computing, Relational Database Systems, SQL, JDBC, Prepared Statement, Callable Statement.

UNIT5: Enterprise programming (11 L)

Java Beans: The software component assembly model- The java beans development kit- developing beans JAR files-Introspection-Bound Properties-Persistence-customizers - java beans API. EJB: EJB architecture- EJB requirements –EJB session beans- EJB entity beans-EJB Clients.

Java Servlet: Servlet overview, Brief origin and advantages over CGI, Writing small Servlet Programs, Deployment Descriptor, Servlet Life Cycle, Sharing Information, Initializing a Servlet, Writing Service Methods, Filtering Requests and Responses, Invoking Other Web Resources, Accessing the Web Context, Maintaining Client State, Finalizing a Servlet, Session: Definition, Different ways to track sessions.

JSP: Introduction to JSP, JSP processing, JSP Application Design, Tomcat Server, Implicit JSP objects, Conditional Processing, Declaring variables and methods, Error Handling and Debugging, Sharing data between JSP pages- Sharing Session and Application Data. Accessing a database from a JSP page, Application-specific Database Action, Developing Java Beans in a JSP page, introduction to Struts framework.

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

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

List of Experiments

S. No.	EXPERIMENT NAME
1	WAP in java to perform basic arithmetic operations.
2	WAP in java to create database connectivity with organization database & display the record of employee table.
3	WAP to implement Runnable Interface.
4	WAP to create 5 thread by extending thread class.
5	WAP to create table Register in database and perform navigation operation using Result Set.
6	WAP in JDBC to perform Transaction Management by using Set Auto Commit (), Commit () & rollback () in the table student.
7	WAP in JDBC to insert Organization details by using Prepared Statement Interface.
8	WAP to demonstrate Life Cycle of Servlet.
9	WAP to switch ON and OFF the bulb by using visible bean.
10	Create a java application by implementing Generic Servlet Class.
11	WAP to display the utilization of Http Servlet Class.
12	Create login application in java by using MVC and mysql.
13	WAP using JSP code for describing all types of directive.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Computer Organization and Architecture
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. The course will consist of assigned reading, weekly lectures, a midterm and final exam, and a sequence of assignments. The goal of the readings and lectures is to introduce the core concepts. The goal of the assignments is to give students some exposure to operating system code. Students are expected to read the course materials prior to each class, and to participate in in-class discussions.

COURSE OBJECTIVES

- To understand the services provided by and the design of an operating system.
- To understand the structure and organization of the file system.
- To understand what a process is and how processes are synchronized and scheduled.
- To understand different approaches to memory management.
- Students should be able to use system calls for managing processes, memory and the file system.

COURSE OUTCOMES

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

CO1: Learn the general architecture& functioning of computers with operating system.

CO2: Describe, contrast and compare differing structures for operating systems.

CO3: Understand and analyze theory and implementation of: processes, resource control (concurrency etc.).

CO4: Understands physical and virtual memory, scheduling, I/O and files

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 Systems & 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", PearsonEducation ,1st Edition 2009.
2.   D M Dhamdhere, "Operating Systems: A Concept based Approach", PHI. 3rd Edition.2017.

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

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> FIFOb>LRUc>LFU

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Discrete Mathematics
7.	Course Basket	Discipline Core

COURSE SUMMARY

The students will learn the basic theory of database. They will be able to design and develop database using conceptual schema, logical schema and physical schema and are expect 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) to the efficient functioning of an organization. This course covers theory and practice in designing a relational database management system with 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: Apply and create Relational Database Design process with Normalization and De-normalization of data.

CO2: Analyze and design database using E-R data model by identifying entities, attributes and relationships.

CO3: To work on MySQL database management system.

CO4: To create database and query the database for information retrieval.

CO5: To design a database so that data redundancy, data inconsistency and data loss problems may be resolved. Understand and implement Graphs and Graph Algorithms.

CURRICULUM CONTENT

Unit 1: Introduction to Database System (7 L)

Introduction: Data base System Applications, data base System VS file System, Data Abstraction, Instances and Schemas, data Models: the ER Model, Relational Model & Other Models , Database Languages, data base Users and Administrator, data base 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 E-R 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 Data base Design, Multi valued 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, Data base Management Systems, TATA McGrawHill 3rd Edition,2014
2. Silberschatz, Korth, Data base System Concepts, McGraw hill, 6th edition,2013
3. ElmasriNavate, Fundamentals of Database Systems, Pearson Education,7th edition 2016

REFERENCES

1. Peter Rob & Carlos Coronel, Data base 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) 12[th]ed.

TEACHING AND LEARNING STRATEGY

The teaching of students will be conduct through power point lectures, tutorials, and short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in Moodle to enable them have appropriate reading.

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 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 table.
7	Creation of various types of JOINS.
8	Implementation of Views and Indices in database.
9	Implementation of foreign key on database.
10	Modify the database structure and drop the record with structure.

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Data Structure & Introduction to Java Programming
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: Methods and tools to design, implement, test, document and maintain a software system.

CO2: Access, retrieve and evaluate relevant professional information reliably.

CO3: Apply empirical and mathematics research methods, technique and problem solving approaches for developing software.

CO4: Be receptive to new ideas and innovation.

CO5: Ability to apply software engineering principles and techniques to develop large-scale software systems. Understand and implement various types of searching, sorting and hashing.

CURRICULUM CONTENT

Unit I   (8 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 II (9 L)

Software Requirement Analysis and Specifications: Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Data Flow Diagrams, Data Dictionaries, Entity-Relationship diagrams, Software Requirement and Specifications, Functional and non-Functional requirements, Software Prototyping, Feasibility Study, Information Modeling, Decision Tables, SRS Document, IEEE Standards for SRS, Software Quality Assurance (SQA),SEI-CMM Model.

Unit III (9 L)

Design: Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead"s Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.

Unit IV (10 L)

Software Reliability: Failure and Faults, Reliability Models: Basic Model, Logarithmic Poisson Model, Calendar time Component, Reliability Allocation. Coding: Top-Down and Bottom –Up programming, structured programming, Compliance with Design and Coding Standards.
Software Project Management:Project planning and Project scheduling. Software Metrics: Size Metrics like LOC, Token Count, Function Count. Cost estimation using models like COCOMO. Risk management activities. Software Reliability and Quality Assurance: Reliability issues, Reliability metrics, reliability models, Software quality, ISO 9000 certification for software industry, SEI capability maturity model. Computer-aided software engineering (CASE), software reuse, component-based software development, extreme programming.

Unit V (10 L)

Testing: Objectives, Testing Tools & Standards. Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Top-Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Path Testing, Structural Testing (White Box Testing), Functional Testing (Black Box Testing),
Maintenance: Corrective and Perfective Maintenance, Maintenance Process, Maintenance Models, Maintenance Cost, Software Re-Engineering, Reverse Engineering. Constructive Cost Models (COCOMO).
Software Quality Management: Software Quality Factors, Quality Assurance, Quality Standards, Software Maintenance.

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

Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Discrete Mathematics
7.	Course Basket	Discipline Core

COURSE SUMMARY

The course introduces some fundamental concepts in theory of computations and formal languages, including language, their type, and related grammar, finite automaton, regular expression, formal language, 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.

COURSE OUTCOMES

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

CO1. Analyze and design finite automata, pushdown automata, Turing machines, formal languages, and grammars.

CO2. Demonstrate their understanding of key notions, such as algorithm, computability, decidability, and complexity through problem solving.

CO3. Proof of the basic results of the Theory of Computation.

CO4. Use of Turing machine in computational mathematics

CO5. Use of automata theory in application 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 (7 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.

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.

TEXTBOOK(S)

1. Hopcroft H.E. and Ullman J. D ,"Introduction to Automata Theory Language and Computation",., Pearson Education.3rd Edition.2006.

2. J. C. Martin, "Introduction to Languages and the Theory of Computation", 4th edition, Tata McGraw-Hill.2010.

3. K.L.P. Mishra, "Theory of Computer Science", PHI.3rd Edition 2014.

REFERENCE BOOKS

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

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Theory of Computation
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 -

CO1: Find out the tokens from high level language program and also understand the working of lexical analyzer.

CO2: Understand Different type of parsing techniques and will be able to differentiate between them.

CO3: Understand the importance of semantic rules.

CO4: Understand importance of symbol table and error handling while compilation process.

CO5: Student should analyze the program and minimize the code by using optimizing techniques which helps in reducing the no. of instructions in a program and also utilization of registers in an effective way.

CURRICULUM CONTENT

Unit I: 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 II: 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 III : 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 IV : 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 V: 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 BOOK:

1.   Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, "Compilers-Principles, Techniques & Tools", Pearson Education, 2nd Edition, 2008

REFERENCE BOOK (S):

1. Robin Hunter, "Essence of Compilers", TBS (1998)

2. Steven S. Muchnick, Advanced Compiler Design & Implementation, Morgan Kaufmann Publishers, 1st edition, 1997

TEACHING AND LEARNING STRATEGY

All materials (pdf/ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

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

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Advanced Java Programming
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: The student will develop an ability to describe what a computer network is and how data communication takes place between two computers connected to a network.

CO2: Able to explain the protocol layering architecture and what are the different functions of each layer.

CO3: Able to identify the requirements for network design in terms of hardware and software used in network construction.

CO4: Able to explain the IPV4 addressing technique, including classful & classless address along with subnetting.

CO5: 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: Introduction to Physical Layer, Data and Signals, Analog and Digital Signal, Transmission Impairments, Digital-to-digital conversion, Analog-to-Digital Conversion, Transmission Modes: Parallel and serial, Digital-to-Analog Conversion, Analog-to-Analog Conversion, 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, Stop-and-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.

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

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 having Two subnets and two Routers using DTE/ DCE Cable with user defined clock rate.
5	Simulate a network using Star Topology Using Cisco packet Tracer.
6	Simulate a network using Bus Topology Using Cisco packet Tracer.
7	Simulate a network using Ring Topology Using Cisco packet Tracer.
8	Simulate a network using Mesh Topology Using Cisco packet Trace.
9	Create a DHCP server using Cisco packet tracer
10	Implement Intra domain and Inter domain routing Protocol using Cisco Packet Tracer.
11	Implement Bit Stuffing using Turbo C++ Editor.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	Data Structures
7.	Course Basket	Discipline Core

COURSE SUMMARY

Comprehensive introduction to the study of computer algorithms with its analysis (time and space complexity). Study of various techniques (Divide & Conquer, Greedy, Dynamic Programming, Backtracking, and Branch & Bound) to design an algorithm. Introduction of the problems that comes under category of P and NP.

COURSE OBJECTIVES

This course aims to provide the knowledge and understanding the complexity issues of algorithms
1. To introduce algorithms analysis and design techniques
2. To understand and design of algorithms used for searching, sorting, indexing operation.

COURSE OUTCOMES

CO1. Analyzing complexity issues of algorithms

CO2. Ability in using the appropriate algorithm for searching, sorting, indexing operations

CO3. Designing of new algorithms

CO4. Student will be able to learn 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 its 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 knapsack problem, 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.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

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. Ellis Horowitz, SatrajSahni and Rajasekharam, Fundamentals of Computer Algorithms, Universities Press; Second edition (2008).

2. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, "Introduction to Algorithms", MIT Press; 3rd edition (2009).

3. Anany Levitin, "Introduction to the Design and Analysis of Algorithm", Pearson Education India; 2nd edition (2008).

4. M.T.Goodrich and R.Tomassia, Algorithm Design: Foundations, Analysis and Internet examples, John Wiley & Sons; 1st edition (2001)

REFERENCE BOOKS:

1. R.C.T.Lee, S.S.Tseng, R.C.Chang and T.Tsai, Introduction to Design and Analysis of Algorithms A strategic approach, McGraw-Hill Education (Asia) ,2005

2. Aho, Ullman and Hopcroft ,Design and Analysis of algorithms, Pearson Education India; 1st edition 2002

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

List of Experiments

S.NO.	EXPERIMENT NAME
1	Program in C to Implement Insertion sort, selection sort
2	Program in C to Implement Quick Sort
3	Program in C to Implement Merge Sort
4	Program in C to Implement Binary Searching, Heap sort
5	Program in C to Implement Activity Selection problem
6	Program in C to Implement job scheduling with deadlines
7	Program in C to Implement fractional knapsack problem
8	Program in C to Implement single source shortest path (Dijkstra Algorithm)
9	Program in C to Implement 0-1 Knapsack problem using Dynamic Programming
10	Program in C to Implement all pair shortest path

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	CSE
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)	Design and Analysis of Algorithms
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: Ability to identify and formulate appropriate AI methods for solving a problem.

CO2: Ability to implement AI algorithms.

CO3: Ability to compare different AI algorithms in terms of design issues, computational complexity, and assumptions.

CO4: Student will be able to use the concepts of AI for real world problem solving.

CO5: Visualize the basic use of AI algorithms and their implementations in laboratory.

CURRICULUM CONTENT

Unit I (8 L)

Introduction- Definitions, Intelligent Agents, Problem solving and Search- Uninformed Search, Informed Search, MiniMax Search, Constraint Satisfaction Problem, A*, Best Search, DFS, BFS.

Unit II (7 L)

Prolog-Introduction to Prolog, Syntax and Meanings of Prolog Programs, Operators and Arithmetic, Prolog for Artificial Intelligence.

Unit III (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 (8 L)

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

Unit V   (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 J. 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).

REFERENCE BOOKS:

1. Ivan Bratko, "Prolog Programming for Artificial Intelligence", Addison Wesley; 4th edition (August 2011).
2. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Prentice-Hall (1990)

TEACHING AND LEARNING STRATEGY

The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material (pdf/ppts, assignments, labs, etc.) will be availed to the students in Moodle (lms.dituniversitu.edu.in) to enable them have appropriate reading.

List of Experiments

S. No.	EXPERIMENT NAME
1	Introduction to PROLOG programming, PROLOG platform. "Hello World" program.
2	Defining Clauses and Predicates, Variables, Anonymous Variables.
3	Arithmetic Operators, Arithmetic Functions and Logical Operators (NOT, conjunction disjunction).
4	Binding Variables and Backtracking & Concept of Unification.
5	Implementation of Recursion in PROLOG.
6	Implementation of LIST and built-in predicates of LIST in PROLOG.
7	Implementation of State-Space Searching Problem using PROLOG (Water-Jug or 8 Queens problem).
8	Universal and Existential Quantifier Variables in PROLOG.
9	Knowledge Base and Rule Base Creation for a specific domain in PROLOG.
10	Implementation of Resolution process in PROLOG.
11	Implementation of an Expert System for a particular domain.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	MA202
7.	Course Basket	

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
1. On successful completion of the course, students will be able to achieve the following:
2. Differentiate various learning approaches, and to interpret the concepts of supervised and unsupervised learning.
3. Compare the different dimensionality reduction techniques.
4. Illustrate the working of classifier models like SVM, Neural Networks and identify classifier model for typical machine learning applications.
5. Apply clustering algorithms and identify its applicability in real life problems.

Curriculum Content

Unit 1: Introduction [5]

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:
Regression & Clustering [5]
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: Classification [5]

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: Dimensionality Reduction [5]

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: Neural Networks [6]

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, 2013.
2. Ethem Alpaydin, Introduction to Machine Learning, PHI, 3rd edition, 2015.

Reference Books

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.

List of Experiments:

s. No.	Title of experiment
1	**Exp 1: Environment Set-up** • Download set-up files • Installation • Environment Set-up • Demo Program
2	**Exp 2: Data Handling** • Downloading Dataset • Import/Export Dataset Files • Summarization of dataset
3	**Exp 3: Data Pre-processing** • Missing Value • Outlier Handling • Formatting data
4	**Exp 4: Descriptive Statistics** a. variance, standard deviation, shape – skewness, kurtosis, percentiles, five point summary

	b. boxplots, histograms, bar plot, pie chart, scatter plot, two way tables, c. covariance, correlation analysis, Chi-Square test for two way tables
5	**Exp 5: Supervised Learning- Regressions** • Linear Regression with one variable • Linear Regression with multiple variable • Polynomial regression
6	**Exp 6: Supervised Learning- Classifications** a. Logistic Regression b. Decision Tree c. k-Nearest Neighbours d. Support Vector Machine
7	**Exp 7: Unsupervised Learning- Clustering** a. K-means clustering b. Hierarchical clustering
8	**Exp 8: Deep Learning** a. Implementation of ANN b. Implementation of CNN c. Implementation of RNN
9	**Exp 9: Data Transfer Learning** a. Importing pre-trained models b. Implementation of pre-trained models with a new learning model

Tools/Software for experiments: Python / R Programming Language

Teaching and Learning Strategy

All materials (ppts, assignments, lab's manual, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Benchmarking:

2.   **IIT Kharagpur** (https://cse.iitkgp.ac.in/~saptarshi/courses/ml2018spring/)

3.   **VNIT, Nagpur** (http://cse.vnit.ac.in/people/rbkeskar/teaching-2/current-courses/machine-learning/)

4.   **Illinois Institute of Technology, Chicago** (http://www.cs.iit.edu/~agam/cs584/#syllabus)

5.   **Massachusetts Institute of Technology, Massachusetts** (http://lcsl.mit.edu/courses/ml/1718/)

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Course Description Document

Department offering the course	CSE
Course Code	CSF342
Course Title	Fuzzy Logic and Neural Network
Credits (L:T:P:C)	2:0:1:3
Contact Hours (L:T:P)	2:0:2
Prerequisites (if any)	
Course Basket	Discipline Elective

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

Course Outcome:
On successful completion of this course, students will learn the following.
1. This will help students to understand the concepts of Fuzzy Logic.
2. This will also help the students to apply different fuzzy operations and functions.
3. Students will be able to learn the concepts of Neural Network.
4. Students will also be able to understand and implement different Activation Functions.
5. Students will be able to implement different Neural Network techniques on different real-life problems.

Curriculum Content

Unit I   [3]
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 II   [6]
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 III   [6]
Introduction to Neural Networks: Introduction to Biological Neural Network, Artificial Neural Network. Activation Functions, Basic Learning rules, Hebb's rule, Biases and Thresholds, Perceptron, Convergence Theorem, Delta Rule, Hyperparameter, Cost Function, Applications of Artificial Neural Networks.

Unit IV   [5]
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 V [6]

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

Suggested Books:

1. Simon S. Hayk in, Neural Networks, Prentice Hall, 3rd edition, 2009.
2. Zimmermann, Fuzzy Set theory and its application", 4th edition, 2016.

Teaching and Learning Strategy

All materials (ppts, assignments, lab's manual, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

LAB EXPERIMENTS (USING MATLAB)

1. To understand MATLAB environment by exploring each option.
2. Write Down A Program to Perform Basic Matrix Operations.
3. Write A MATLAB Program to Plot A Few Activation Functions That Are Being Used in Neural Networks
4. Generate And not Function Using Mcculloch-Pitts Neural Net by A MATLAB Program.
5. With A Suitable Example Simulate the Perceptron Learning Network and Separate the Boundaries. Plot the Points Assumed Respective Quadrants Using Different Symbols for Identification.
6. With A Suitable Example Demonstrate the Perceptron Learning Law with Its Decision Regions Using MATLAB. Give the Output in Graphical Form
7. Write A MATLAB Program to Illustrate Art Neural Network
8. Write Down A Program to Implement Perceptron Learning Rule for Linearly Separable Problem
9. Write Down A Program to Implement Perceptron Learning Rule for Non-Linearly Separable Problem
10. Write Down A Program to Draw A Graph with Multiple Curve
11. Write Down A Program to Implement Fuzzy Associative Memory.
12. Write Down A Program to Implement Hopfield Neural Network.

BENCHMARK

- https://cse.iitkgp.ac.in/~dsamanta/courses/sca/index.html

- https://nptel.ac.in/content/syllabus_pdf/127105006.pdf
- https://www.seas.harvard.edu/courses/cs281/files/syllabus.pdf

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	Discipline Elective

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

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

1. Understand the fundamental of evolution based learning algorithms, advanced searching and optimization techniques.
2. Understand and implement the concepts of genetic algorithms and genetic programming
3. Ability to solve problems using swarm intelligence, Ant Colony Optimization
4. Understand multimodal problems and their solution and understand the concept of Spatial Distribution
5. Visualize the basic use of evolutionary computing algorithms and its implementations with neural network and parallel Evolutionary Computing.

Curriculum Content

Unit 1: EVOLUTIONARY COMPUTING BASICS [5]

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: GENETIC ALGORITHMS & GENETIC PROGRAMMING [5]

GENETIC ALGORITHMS

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, Initialisation, Bloat in Genetic Programming, Problems Involving ``Physical'' Environments, Example Applications.

Unit 3: SWARM INTELLIGENCE [5]

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.

Unit4: MULTIMODAL PROBLEMS AND SPATIAL DISTRIBUTION [5]

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

Unit- 5: ADVANCED TOPICS [6]

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.

Reference Books

1- Kenneth A. De Jong, "*Evolutionary Computation, A unified Approach* ", The MIT Press Cambridge, Massachusetts London, England, 1st edision, 2006.

2- S. Sumathi & T.Hamsapriya & P.Surekha, "*Evolutionary Intelligence-An Introduction to theory and applications with Matlab*". Springer-Verlag Berlin Heidelberg, 3rd edision, 2008.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Teaching and Learning Strategy

All materials (ppts, assignments, lab's manual, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

S. No.	Programme
1	**Write A Program For Implementing Linear Saturating Function.**
2	**Write A Program For Error Back Propagation Algorithm (Ebpa) Learning.**
3	**Study And Analysis Of Genetic Algorithm Life Cycle.**
4	**Study And Analysis Of Fuzzy Vs Crisp Logic.**
5	**Write A Program Of Perceptron Training Algorithm.**
6	**Write A Program To Implement Hebb's Rule**
7	**Write A Program To Implement Of Delta Rule**
8	**Write A Program For Back Propagation Algorithm**

Course Description Document

1.	Department offering the course	CSE
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)	
7.	Course Basket	Discipline Elective

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 Objective: The course will start with a brief introduction to robots and robotics. Various methods of robot teaching will be explained with some suitable examples. The working principles of various sensors used in robots will be explained in detail. The principles of motion planning algorithms will be explained in detail.

Course Outcome: At the end of the course, the student will learn the following:
1. Learn how to keep robots in modern industries.
2. Learn how to apply robots in different areas (space, medical, manufacturing etc.).
3. Students will have brief idea of different components of robots system and their working principle.
4. Understand the notion of an agent, how agents are distinct from other software.
5. Understand the key issues in designing societies of agents that can effectively cooperate in order to solve problems, including an understanding of the key types of multi-agent interactions possible in such systems.
6. Understand the main application areas of agent-based solutions, and be able to develop a meaningful agent-based system using a contemporary agent development platform.

Curriculum Content

Unit I: Introduction to Robotics: [5]
Components, Classification, Characteristics and Applications of Robots. Material transfer, Machine loading/unloading, Processing operation, Assembly and Inspection, Feature Application.

Unit II: Drive Systems, Actuators and Sensors: [5]
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: Basic Concepts of Artificial Intelligence: [5]
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: Intelligent Agents and Multi-agent system: **[6]**

Design of intelligent agents - reasoning agents, agents as reactive systems; hybrid agents; layered agents. Multi-Agent Systems: Classifying multi-agent interactions - cooperative versus non-cooperative; zero-sum and other interactions; Cooperation, Prisoner's dilemma and Axelrod's experiments; Interactions between self-interested agents: auctions & voting systems: negotiation; Interactions between benevolent agents: cooperative distributed problem solving, partial global planning; coherence and coordination; Interaction languages and protocols: speech acts, KQML/KIF, the FIPA framework.

Unit 5: Multi-robot representations and Task Planning: **[6]**

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.

Suggested Books:

1. Introduction to Robotics Analysis, Systems, Applications by Saeed B Niku, Prentice Hall, 2nd edition, 2001.
2. Fundamentals of Robotics Analysis and Control, Robert J Schilling, PHI, 5th edision, 2003.
3. An Introduction to MultiAgent Systems - Second Edition. Michael Wooldridge Wiley, latest edition.
4. Mikell P Groover, Nicholas G Odrey, Mitchel Weiss, Roger N Nagel, Ashish Dutta, "Industrial Robotics, Technology programming and Applications", McGraw Hill, 2nd edision, 2017.
5. Craig. J. J. "Introduction to Robotics- mechanics and control", Addison- Wesley, 3rd edition, 2004.

Teaching and Learning Strategy

The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in Moodle to enable them have appropriate reading.

BENCHMARK

1.   University of Minnesota:
https://www.cs.umn.edu/research/research_areas/robotics-and-artificial-intelligence
2.   University of Liverpool:
https://intranet.csc.liv.ac.uk/teaching/modules/module.php?code=COMP310
3.   Gujarat Technological University: http://gtu.ac.in/syllabus/NEW%20ME/Sem2/2725001.pdf

LIST OF EXPERIMENTS:

1.   Develop a working robot using a V-rep tool
2.   To develop the programs for (1) Water Jug Problem, (2) Knights Tour, (3) Crypto Arithmetic Problems & (4) Implementing Searching algorithm.
3.   Inter-robot communication.
4.   Speed control of a robot.
5.   Grabbing an object by a robot.
6.   Robot carrying a heavy object from location l1 to l2.
7.   Multiple robots carrying a heavy object from location l1 to l2.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>Undergraduate Course Description Document</u>

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	

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

- Understand the concept of artificial neural networks, convolutional neural networks, and recurrent neural networks
- Discuss how to speed up neural networks along with regularization techniques to reduce overfitting
- Understand the concept of generative models
- Implement deep learning algorithms, and learn how to train deep networks

Curriculum Content

UNIT 1: BASICS [5]
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]
Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, autoencoders.

UNIT 3 : FEED FORWARD NETWORKS [5]
Deep Neural Networks: Difficulty of training deep neural networks, Greedy layerwise 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]**

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]**

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.

Textbook(s)

1. Deep Learning", Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016.

Reference Books

2. "Neural Networks: A Systematic Introduction", Raúl Rojas, 1996 2. Pattern Recognition and Machine Learning, Christopher Bishop, 2007

List of Experiments:

s. No.	Title of experiment
1	**Exp 1: Data Handling** • Downloading Dataset • Import/Export Dataset Files Summarization of dataset
3	**Exp 2: Supervised Learning- Regressions** • Linear Regression with one variable • Linear Regression with multiple variable Polynomial regression
4	**Exp 3: Supervised Learning- Classifications** e. Logistic Regression f. Decision Tree g. k-Nearest Neighbours h. Support Vector Machine
5	**Exp 4: Unsupervised Learning- Clustering** c. K-means clustering d. Hierarchical clustering
6	**Exp 5: Deep Learning** d. Implementation of ANN e. Implementation of CNN f. Implementation of RNN
7	**Exp 6: Data Transfer Learning** c. Importing pre-trained models d. Implementation of pre-trained models with a new learning model

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Tools/Software for experiments: Python / R Programming Language

Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Benchmarking:
IIT Kharagpur (https://cse.iitkgp.ac.in/~saptarshi/courses/ml2018spring/**)**
Illinois Institute of Technology, Chicago (http://www.cs.iit.edu/~agam/cs584/#syllabus **Massachusetts**
Institute of Technology, Massachusetts (http://lcsl.mit.edu/courses/ml/1718/**)**

Track: Data Science and Analytics

S. No.	Course Title (Proposed)	Credits: L T P C
1.	Introduction to Data Science	2 0 1 3
2.	Data Mining and Data Warehousing	2 1 0 3
3.	Machine Learning	2 0 1 3
4	Big Data Analytics	2 0 1 3
5.	Deep Learning	2 0 1 3
6.	R Programming	2 0 1 3

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	**Department offering the course**	**Computer Science and Engineering**
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)**	
7.	**Course Basket**	**Discipline Elective**

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

- An understanding of problems solvable with data science and an ability to attack them from a statistical perspective.
- An understanding of when to use supervised and unsupervised statistical learning methods on labeled and unlabeled data-rich problems.
- The ability to create data analytical pipelines and applications in Python.
- Familiarity with the Python data science ecosystem and the various tools needed to continue developing as a data scientist.

Curriculum Content

UNIT1: COMPUTER SCIENCE/STATISTICS/LINEAR ALGEBRA SHORT REVIEW
[5]

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.

UNIT2: EXPLORATORY DATA ANALYSIS (EDA) AND VISUALIZATION   [5]
Exploration of dataset using Pandas Python package. Compute Stem and Leaf plot, Box plot, Histogram, Scatter plot of a data set. At this point, students decide on a course project that would benefit from the data-scientific approach. The project involves exploration of freely-accessible and usable data that answer some interesting questions about the data (Several resources of free data will be provided.)

UNIT3: DATA MODELING: SUPERVISED/UNSUPERVISED LEARNING [5]

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.

UNIT4: DATA MODELING: FEATURE SELECTION, ENGINEERING, AND DATA PIPELINES [5]

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.

UNIT5: DATA MODELING: MODEL EVALUATION AND PROJECT PRESENTATIONS [6]

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

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

List of Experiments:

S. No.	Title of experiment
1	

Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>Undergraduate Course Description Document</u>

1.	Department offering the course	Computer Science and Engineering
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)	
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 covers 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

Upon successful completion of the course, the student will be able to:

- Explain and evaluate the various data mining algorithms
- Discover and measure interesting patterns from different kinds of databases.
- Apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data.

Curriculum Content

UNIT I: OVERVIEW: 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.
[5]

UNIT II DATA PRE-PROCESSING: 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. [5]

UNIT III CONCEPT DESCRIPTION: 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 [5]

UNIT IV CLASSIFICATION: What is Classification, Issues regarding Classification, Decision tree, Bayesian Classification, Classification by Back propagation.   [5]

UNIT V CLUSTER ANALYSIS: Data types in cluster analysis, Partitioning methods. Hierarchical Clustering- CURE and Chameleon, Density Based Methods-DBSCAN, OPTICS, Grid Based Methods-STING, CLIQUE, Outlier Analysis   [6]

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

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

List of Experiments:

s. No.	Title of experiment
1	Build Data Warehouse and Explore WEKA . List all the categorical (or nominal) attributes and the real-valued attributes separately. What attributes do you think might be crucial in making the credit assessment?
2	Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets.
3	Demonstrate performing classification on data sets.
4	Demonstrate performing clustering on data sets.
5	Demonstrate performing Regression on data sets.
6	Task 1: Credit Risk Assessment. Sample Programs using German Credit Data.
7	Task 2: Sample Programs using Hospital Management System -Simple Project on Data Preprocessing.

Tools/Software for experiments: Python / R Programming Language

Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	

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 Data Science 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 studying this course, students will be able to:

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

2. Use the R programming for preprocessing the real-life datasets.

3. Understand and analyze the descriptive statistics for a given dataset.

4. Implement some classical machine learning models using R programming.

Curriculum Content

UNIT 1: INTRODUCTION TO R [5]
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: R PROGRAMMING CONSTRUCTS [5]
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: DATA MANAGEMENT [5]
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: DESCRIPTIVE STATISTICS USING R   [5]
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: R FOR MACHINE LEARNING   [6]
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.   Practical Data Science with R. Author(s): Nina Zumel, John Mount, Manning Shelter Island, 2014.
2.   Beginning R: The statistical programming language. Mark Gardener, John Wiley & Sons, 2012

Reference Books

1.   R programming for data science. R. D. Peng, Leanpub, 2016.
2.   The R book, Crawley and Michael, John Wiley & Sons, 2012.

List of Experiments:

s. No.	Title of experiment
1	R Environment Set-up: Download set-up files, Installation, Environment Set-up, Demo Program
2	R Objects: Object, Vector, List, Matrix, Array, Data Frame and Manipulating Objects.
3	R Data Handling: Downloading Dataset, Import/Export Dataset Files, Summarization of dataset
4	Data Pre-processing using R: Missing Value, Outlier Handling, Formatting data
5	Descriptive Statistics using R: variance, standard deviation, shape – skewness, kurtosis, percentiles, five point summary: boxplots, histograms, bar plot, pie chart, scatter plot.
6	Descriptive Statistics using R: covariance, correlation analysis, Chi-Square test for two way tables.
7	Supervised Learning- Regressions using R: Linear Regression with one variable, Linear Regression with multiple variable, Polynomial regression
8	Supervised Learning- Classifications using R Logistic Regression, Decision Tree, k-Nearest Neighbors, Support Vector Machine
9	Unsupervised Learning- Clustering using R: K-means clustering, Hierarchical clustering

Tools/Software for experiments: R studio

Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Benchmarking:

1. Introduction to R software, IIT Kanpur (NPTEL), (https://nptel.ac.in/courses/111104100/#)
2. IIIT Bangalore

(https://www.iiitb.ac.in/pg-diploma-data-science)

3. R Programming, John Hopkins University (Coursera)

(https://www.coursera.org/learn/r-programming.

4. Data Science for Engineers, IIT Madras (NPTEL) (https://nptel.ac.in/courses/106106179/

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>Undergraduate Course Description Document</u>

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	

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
- Understand the need and process of data analysis.
- Learn the different component of Hadoop Ecosystem.
- Understand the Map Reduce and the use of Apriori and Fp-Growth.
- Learn to analyse the data using R.
- Analyse different software for processing Big Data.

Curriculum Content

UNIT 1: INTRODUCTION TO BIG DATA AND HADOOP [5]

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) [5]

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 [5]
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 [5]
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   **[6]**

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

Textbook(s)

1.   **Intelligent Data Analysis**, Michael Berthold, David J. Hand,2/e, Springer, 2015.
2.   **Mining of Massive Datasets**, Anand Raja Raman and Jeffrey David Ullman,2/e, Cambridge University Press, 2012.
3.   **Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics,** Bill Franks, 1/e, John Wiley & sons, 2012.
4.   **Hadoop: The Definitive Guide**, Tom White Third Edition, O'reillyMedia,2012.

Reference Books

1.   **Making Sense of Data,** I, Glenn J. Myatt, 2/e, John Wiley & Sons, 2014
2.   **Big Data Glossary**, Pete Warden,1/e, O'Reilly, 2011.
3.   **Data Mining Concepts and Techniques**, Jiawei Han, Micheline kamber, 2/e, Elsevier, Reprinted 2015.

List of Experiments:

s. No.	Title of experiment
1	Installation of Hadoop.
2	Directory Management Tasks in Hadoop a. Create a directory in HDFS b. List the Contents of directory c. Remove a directory in HDFS
3	File Management Tasks in Hadoop a. Upload and download a file in HDFS b. See Contents of a File. c. Remove a file in HDFS.
4	File Transfer in Hadoop a. Copy a file from Source to destination. b. Move file from Source to Destination.
5	Word Count Map Reduce program to understand MAP Reduce Paradigm.
6	Weather Report POC-Map Reduce Program to analyse time-temperature statistics and generate report with max/min temperature.
7	Implementing Matrix Multiplication with Hadoop Map Reduce.
8	Pig, Latin Scripts to sort, Group, Join Project and Filter the data.

9	Introduction to Weka tool to process data.
10	Use R to process data and visualize it using ggplot2

Tools/Software for experiments: Hadoop

Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Benchmarking:
1.   **Columbia University, New York (**https://www.ee.columbia.edu/~cylin/course/bigdata/ **)**
2.   **The Graduate Institute Geneva**
(https://www.karstendonnay.net/download/spring2018/Syllabus_MINT-078.pdf)
3.   **NSUT Delhi**
(http://www.nsit.ac.in/static/documents/IS.pdf)
4.   **IIIT Delhi**
(https://www.iiitd.ac.in/academics/courses/institute#CSE510A)

Undergraduate Course Description Document

8.	Department offering the course	Computer Science and Engineering
9.	Course Code	CSF441
10.	Course Title	DEEP LEARNING
11.	Credits (L:T:P:C)	2:0:1:3
12.	Contact Hours (L:T:P)	2:0:2
13.	Prerequisites (if any)	
14.	Course Basket	

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

- Understand the concept of artificial neural networks, convolutional neural networks, and recurrent neural networks
- Discuss how to speed up neural networks along with regularization techniques to reduce overfitting
- Understand the concept of generative models
- Implement deep learning algorithms, and learn how to train deep networks

Curriculum Content

UNIT 1: BASICS [5]
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]
Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, autoencoders.

UNIT 3 : FEED FORWARD NETWORKS [5]
Deep Neural Networks: Difficulty of training deep neural networks, Greedy layerwise 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]
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]

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.

Textbook(s)

Deep Learning", Ian Goodfellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016.

Reference Books

"Neural Networks: A Systematic Introduction", Raúl Rojas, 1996 2. Pattern Recognition and Machine Learning, Christopher Bishop, 2007

List of Experiments:

s. No.	Title of experiment
1	**Exp 1: Data Handling** • Downloading Dataset • Import/Export Dataset Files Summarization of dataset
3	**Exp 2: Supervised Learning- Regressions** • Linear Regression with one variable • Linear Regression with multiple variable Polynomial regression
4	**Exp 3: Supervised Learning- Classifications** i. Logistic Regression j. Decision Tree k. k-Nearest Neighbours l. Support Vector Machine
5	**Exp 4: Unsupervised Learning- Clustering** e. K-means clustering f. Hierarchical clustering
6	**Exp 5: Deep Learning** g. Implementation of ANN h. Implementation of CNN i. Implementation of RNN
7	**Exp 6: Data Transfer Learning** e. Importing pre-trained models f. Implementation of pre-trained models with a new learning model

Tools/Software for experiments: Python / R Programming Language

Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Benchmarking:
IIT Kharagpur (https://cse.iitkgp.ac.in/~saptarshi/courses/ml2018spring/)
Illinois Institute of Technology, Chicago (http://www.cs.iit.edu/~agam/cs584/#syllabus **Massachusetts**
Institute of Technology, Massachusetts (http://lcsl.mit.edu/courses/ml/1718/)

Internet of Things and Edge Computing

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

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>**Undergraduate Course Description Document**</u>

1.	Department offering the course	Computer Science and Engineering
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)	CS 203
7.	Course Basket	Discipline Elective

Course Summary

The course is designed for the smooth and fast data transportation over the wired and wireless medium. 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. Number of advance network protocol is included in this course to understand the various network challenges for wired as well as the wireless medium.

Course Objectives

This course is designed to provide knowledge about the basic concepts of Advance Computer Network. The Course includes switching concept for packet forwarding, wireless security, and smooth and secure data transmission over the wired and wireless medium. The course includes the queuing model and Markovian theory for smooth data transportation. The main purpose of this course is to help the student to understand the basic concept about the secure and smooth transportation of data, which is a real world challenging problem.

Course Outcomes

At the end of the course the student should be able to:

1. Understand and explain the basic concepts of Advance Computer Network.
2. Describe the real-life applications based on the fundamental concepts.
3. Understand how to handle the multi state process for fast data transportation over any medium.
4. Understand how to design the secure 'wired and wireless' network.

Curriculum Content
Unit 1: Network Design
Design Principles, Determining Requirements, Analysing the Existing Network, Preparing the Preliminary Design, Completing the Final Design Development, Deploying the Network, Monitoring and Redesigning, Maintaining, Design Documentation, Cisco PDIOO Model, Modular Network Design, Hierarchical Network Design, The Cisco Enterprise Composite Network Model.
(8L)

Unit 2: Switching and routing
Switching Design: Switching Types, Layer 2 and 3 Switching, Multilayer Switching, Cisco Express Forwarding, Switching Security, Multi-Protocol Label, Switching (MPLS), MPLS Architecture and related protocols..IPv4 Routing Design: IPv4 Address Design, Private and Public Addresses, NAT,

Subnet Masks, Hierarchical IP Address Design, Deploying IPv6 in Campus Networks. Router Design: Configuring a Router, Routing Protocols. (8L)

Unit 3: Wireless LAN Design
Wireless Technology Overview, Wireless Standards, Wireless Components, Wireless Security, Wireless Security Issues, Wireless Threat Mitigation, Wireless Management, Wireless Design Considerations, IEEE 802.11, Wireless Standard, Cellular Networks, Mobile IP, Wireless Mesh Networks(WMNs), QoS Models: IntServ, DiffServ, QoS Tools, Policing and Shaping, Congestion Avoidance, Congestion Management, Link- Specific Tools, QoS Design Guidelines.

(8L)

Unit 4: Stochastic Processes: & Queuing System
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/∞ Queue, The M/M/m Queue. (8L)

Unit 5: 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.

(8L)

Textbook(s)
1.	Diane Tiare and Catherine Paquet, "Campus Network Design Fundamentals", Pearson Education, 2006.
2.	Arnold O. Allen, " Probability Statistics, and Queuing Theory with Computer Science Application," Academic Press, Inc.6277 Sea Harbor Drive Orlando, FL United States, 2nd Edition
3.	Gunter Bolch, Stefan Greiner, Hermann de Meer, Kishor S. Trivedi, " Queueing Networks and Markov Chains" John Wiley & Sons, Inc., Publication, 2nd Edition.

Reference Books
5. Craig Zacker, "The Complete Reference: Upgrading and Troubleshooting Networks", Tata McGraw-Hill, 2000 Edition.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

<u>Undergraduate Course Description Document</u>

1.	Department offering the course	Computer Science and Engineering
2.	Course Code	CSF349
3.	Course Title	Cloud Computing
4.	Credits (L:T:P:C)	2:0:1:3
5.	Contact Hours (L:T:P)	2:0:2
6.	Prerequisites (if any)	
7.	Course Basket	Discipline Elective

Course Summary

This course is designed to provide an introduction to cloud computing and virtualization technologies. It provides the basic knowledge of public cloud, private cloud and multi cloud. it will also cover the security perspective/ challenges in the cloud environment.

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

On successful completion of the course, students will be able to achieve the following:
1. Define cloud computing its service and deployment models.
2. Describe the importance of virtualization, multi-tenancy in the cloud environment.
3. Use and Examine different cloud computing services.
4. Describe the different security threats and challenges faced by cloud provider.
5. Differentiate the different types of business cloud and its uses.

Curriculum Content

Unit 1: (3L)
Overview of cloud 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.

Unit 2: (6L)
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: (6L)
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: Multi-Cloud   (6L)

Concept of multi-cloud management, Challenges in managing heterogeneous clouds, benefits of multicloud 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: Cloud Security   (3L)

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. Raj Kumar Buyya, James Broberg, AndrezeiM.Goscinski Cloud Computing: Principles and paradigms. John Wiley & Sons, (2011),
2. Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter (2010) Cloud Computing: A Practical Approach, McGraw Hill.

Reference Books

1. Sosinsky, Barrie. Cloud computing bible. John Wiley & Sons, 2010.
2. Kailash Jayaswal,. Cloud Computing Black Book. John Wiley & Sons, 2014.

List of Experiments:

s. No.	Title of experiment
1	**Exp 1:** Demonstrate the virtualization by enabling the OS virtualization on single machine by creating instances.
2	**Exp 2:** Installation of VM Ware/ virtual box and implement multiple OS.
3	**Exp 3:** Demonstration and Working of Microsoft Hyper-V.
4	**Exp 4:** Install Google App Engine. Create hello world app and other simple web applications using python/java.
5	**Exp 5:** Installation and demonstration of Xen bare metal hypervisor/ KVM Cloud Platform on workstation
6	**Exp 6:** Implementation and demonstration of VPN setup on the windows environment.

7	**Exp 7:** Implementation of Open stack environment and designing of Cloud based migration solution Architecture
8	**Exp 8:** Installation and configuration of Hadoop / Eucalyptus.
9	**Exp 9:** Case study on Amazon EC2/Microsoft Azure/Google Cloud Platform.

Tools/Software for experiments: Python, KVM, XEN, Virtual box, VMWare, open stack, Google app engine.

Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Benchmarking:

5. **IIT Kharagpur** (https://cse.iitkgp.ac.in/?syllabus.html)
6. **NIT Warangal** (https://www.nitw.ac.in/media/Scheme&Syllabus-2014/BTECH-CSE-Syllabus.pdf)
7. **IITDelhi** (http://www.cse.iitd.ernet.in/cse/newcurriculumcontents/newcourses.html#COL733)
8. **National University of Singapore** (http://www.comp.nus.edu.sg/~teoym/CP3109/L0-Overview.pdf)

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	Discipline Elective

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

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. (3L)

Unit 2: Types of Radio Waves, Propagation Mechanisms, Free Space Propagation, Land Propagation, Path loss and Fading, Doppler Effect, Delay Spread and Intersymbol Interference.   (3L)

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.   (6L)

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.

 Reference Books

1. Wireless Communications: Principles and Practice, 2e, by T.S. Rappaport.

Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	**Department offering the course**	**Computer Science and Engineering**
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)**	**Computer Networks**
7.	**Course Basket**	**Discipline Elective**

8.   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.

9.   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.

10.   Course Outcomes:

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.

11.   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.   (12L)
Unit 3: Data transmission in Cloud, IoT Analytics and Visualization   (6L)
Unit 4: IoT Security, IoT Project execution and demonstration   (6L)

LAB CONTENTS:

- Lab 1 and 2: Develop prototype using Arduino Uno,
- Lab 3, 4, 5: Setup and Develop prototype using Raspberry Pi and environmental sensors
- Lab 6, 7, 8, 9: Temperature, Humidity (DHT22), Movement detection and distance measurement using RPI, Detect movement using an HC-SR501 passive infrared (PIR) motion detector, and HC-SR04 ultrasonic ranging module.
- Lab 10-11: Cloud interfacing with SOC devices, sample data upload for visualization and Analysis.
- Lab 12, 13, 14: End-to-End IoT Projects integration, testing and demo.

12.   Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to the modules in Moodle for details.

13.   Bibliography
Textbook: No Textbook. Instructors will provide reading materials.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	Introduction to Java Programming
7.	Course Basket	Discipline Elective

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

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.   (6L)

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.   (6L)

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.   (6L)

Textbook(s)
Head First Android Development: A Brain-Friendly Guide, by Dawn Griffiths and David Griffiths.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

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

Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Teaching of students will be conducted through power point lectures, tutorials, short classroom exercises.

Laboratories:

1. Write an application that has two Activities and two related screens. Name the Activities ActivityScreen1 and ActivityScreen2. Place a button on each screen that allows you to click that button to open the other Activity and related screen.

2. Create an information logging message and add it to the on Create method in any program that you have created. The logging message can say anything you want it to say. After you run the program using the Run Debug Configuration, look for your logging message in the Log Cat section in Eclipse.

3. Open your Graphical User Interface Example application in the emulator, In Eclipse, go to Window and Open Perspective. Open a Hierarchy View. Explore the user interface in the Hierarchy View. What value can this view add in your development? Examine other perspectives discussed in Chapter 4 such as Pixel Perfect and DDMS. Think about how you might use each of these perspectives in your future development projects.

4. Create a new application called Test Layouts. For this application add three string variables in the strings.xml file under resources, and add values that you want to assign to each (such as your own email address and phone number and a website of your choice that you want the user to link to). Create a graphical layout in which all three of these variables are clickable on the display.

5. Using the Test Layouts application created in Lab 4, add an Edit Text field and include a hint for the user about what to type in the field. Add a Spinner control that has seven selection items.

6. Create a new application called Date/Time. Add the Date Picker control to the layout screen. Use the Date Picker. On Date Changed Listener to get the date selected by the user and then display that date in a Text Field.

7. Write a simple program using the Grid Layout and display a box that shows a column span of 3 across the top and bottom, with the top labeled North and the bottom labeled east and which shows the left and right sections labeled East and West.

8. Write a program in which you place a List View within a Linear Layout. Have the list view display seven colors in the list, and display the selection at the bottom of the screen after the selection is clicked. Instead of using the set Content View(int) method, how is the screen displayed?

9. Create a program that will display and allow the user to update the following preferences:
a. Their username (enter into a text field)
b. Welcome Message (enter into a text field)
c. How they want to handle Application Updates (checkbox)
d. Type of data they want to download (from a list selection)

10. Write the code snippet that would allow you to create a text file called "myData" and write the following line of data to this new file "This is my first line of data in a new file." Locate where this file was stored and write the path to that file.

Cybersecurity and Privacy

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

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

<u>UNDERGRADUATE COURSE DESCRIPTION DOCUMENT</u>

1.	Department offering the course	Computer Science and Engineering
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)	Basics of Number Theory
7.	Course Basket	Discipline Elective

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:
1. To introduce the student to elementary number theory, as required for further study of important cryptographic protocols.
2. To introduce the student to the fundamentals of modern symmetric cryptography.
3. 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. Perform 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.

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

LIST OF EXPERIMENTS:

S. No.	Experiment Name
1	Write a program to perform encryption and decryption using the following algorithms a. Caeser cipher b. Generalized Caesar Cipher
2.	Write a program to perform encryption and decryption using the playfair cipher for the following data: Plaintext= CRYPTOGRAPHY, Key= $\begin{pmatrix} 5 & 2 & 11 \\ 3 & 1 & 21 \\ 7 & 0 & 15 \end{pmatrix}$
3	Write a program to perform encryption and decryption using the following algorithms a. Vigenere Cipher b. Affine Cipher
4	Write a program to implement following transposition ciphers: a. Rail Fence Cipher b. Columnar Cipher
5	Write a program to perform show the key generation process for a single round of DES algorithm.
6	Write a program to perform show the working of Euler's Totient Function.
7	Use Extended Euclidean Algorithm and Calculate the Multiplicative inverse.
8	Write a program to show the working of S-DES algorithm for data encryption.
9	Write a program to show the working of Asymmetric Encryption Scheme – RSA.
10	Write a program to show the working of Diffie-Hellman Key Exchange mechanism.

Approved by the Academic Council at its 14th Meeting held on 22.04.2020

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

•	Department offering the course	Computer Science and Engineering
•	Course Code	CSF354
•	Course Title	Data Encryption and Network Security
•	Credits (L:T:P:C)	2:0:1:3
•	Contact Hours (L:T:P)	2:0:2
•	Prerequisites (if any)	Programming experience in C/C++ or JAVA
•	Course Basket	Discipline Elective

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: To Understand the significance of authentication process using digital signature.
CO2: To 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

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

LIST OF EXPERIMENTS:

S.No.	Experiment Name
1	Write a program to show the Implementation of digital signature standard.
2.	Write a program to show the Implementation of RSA digital signature.
3	Write a program to calculate the message digest of a text using the MD5 algorithm.
4	Write a program to calculate the message digest of a text using the SHA-1 algorithm.
5	Perform VTP configuration to establish client/server connection.
6	Demonstration of sniff for router traffic (using any software tool)
7	Implement Caesar Cipher & perform brute force attack on Caesar cipher.
8	Create a Virtual Private Network (VPN) over WAN.
9	Implement and demonstrate IDS using any tool (using any relevant software)
10	Design and implementation of a simple client/server model and running application using sockets and TCP/IP.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	
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: The cyber threat landscape, both in terms of recent developing issues and those issues which persist over time.

CO2: The roles and effects of governments, commercial and other organisations, citizens and criminals in cyber security affairs.

CO3: General values and policies that can be functional to systems to make them more vigorous to attack.

CO4: Key factors in cyber security from different corrective views including computer science, management, law, criminology, and social sciences.

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: Overview, Handling Preliminary Investigations, Controlling an Investigation

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.

TEACHING AND LEARNING STRATEGY

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

UNDERGRADUATE COURSE DESCRIPTION DOCUMENT

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	Cyber Security

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: To understand the concepts of computer forensics fundamentals and types of computer forensics.

CO3: To 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.

TEACHING AND LEARNING STRATEGY
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

LIST OF EXPERIMENTS:

S.No.	Experiment Name
1	Study of Security fundamentals, Ethical Hacking.
2.	Study of System threat attacks - Denial of Services.
3	Study of Sniffing and Spoofing attacks.
4	Study of Different attacks causes by Virus and Trojans
5	Study of Techniques uses for Web Based Password Capturing.
6	Study of Anti-Intrusion Technique – Honey pot.
7	Password Cracking using Rainbow Crack.
8	Extracting Administrator password using LCP.
9	Find hidden files using ADS Spy.
10	Study of FTK Imager, Encase Forensic tools

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

<u>UNDERGRADUATE COURSE DESCRIPTION DOCUMENT</u>

1.	Department offering the course	Computer Science and Engineering
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 (if any)	
7.	Course Basket	Discipline Elective

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: Define the extents affected by cybercrime and investigation.
CO4: Demonstrate tools used in cyber forensic.
CO5: Validate their knowledge of report writing and forensic ethics.

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. Sunit Belapure 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.

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

<u>**UNDERGRADUATE COURSE DESCRIPTION DOCUMENT**</u>

1.	**Department offering the course**	Computer Science and Engineering
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)**	
7.	**Course Basket**	Cyber Security

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 and vulnerabilities in wireless and mobile systems.
CO2: Understand the security mechanisms and authentication procedures.
CO3: Understand the strategies for developing secure mobile applications.
CO4: Use of mobile security penetration tools for evaluating the robustness of mobile applications.
CO5: Gain knowledge about some of the 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 (8 L)

Wi-Fi Security Dedicated Architectures: 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: (7L)

Bluetooth Security: 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.   LeventeButtyán 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.

Computer Vision and Biometrics

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

Course Structure & Syllabus of B.Tech.– Computer Science & Engg. Applicable for Batch: 2020-24

<u>Undergraduate Course Description Document</u>

1.	Department offering the course	Computer Science and Engineering
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)	Basic Mathematics- calculus, probability, statistics; MATLAB/C/C++
7.	Course Basket	Discipline Elective

8.   Course Summary

Biometrics is the science of identifying or authenticating an individual's identity based on behavioural 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.

9.   Course Objectives

The aim is to give students an understanding of biometric systems based on a number of biometric traits such as 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, and hand shape recognition, Biometric system design and performance evaluation, multi-modal biometric systems, and privacy and ethical issues.

10.   Course Outcomes

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

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

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

c.   Understand various Biometric security issues

d.   Describe Cryptography security and Fuzzy models

11.   Curriculum Content

Unit-1   **[9]**

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]**

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]

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]

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

Reference Books

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

12. Teaching and Learning Strategy

The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in Moodle to enable them have appropriate reading.

13. Hardware and Software Requirement

Not Applicable

14. List of Experiments

Not Applicable

Course Structure & Syllabus of B.Tech.– Computer Science & Engg.
Applicable for Batch: 2020-24

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	Basic Mathematics-calculus, linear algebra (Eigen analysis, matrix inverse), Fourier transform and probability). Introduction to Image Processing
7.	Course Basket	Computer Vision & Biometrics

8.   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 includes: Applications in Display Technology, Computer Vision for Navigation, Metrology, High Level Video Analysis, and Human Computer Interfaces.

9.   Course Objectives

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

10.   Course Outcomes

On successful completion of the course, students will be able to achieve the following:
e.   To recognize and identify specific faces among others.
f.   Learn how to install OpenCV and explore basic image processing concepts.
g.   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.
h.   To build a 3D representation of a scene using stereoscopic images.

11.   Curriculum Content

Unit-1   **[4]**
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, modelling complex data densities, regression models, Classification models

Unit-2   **[6]**
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]

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]

Image processing for feature detection and Image synthesis, edge detection, corner detection, line and curve detection, SIFT operator, Image-based modelling 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

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

12. Teaching and Learning Strategy
All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

13. Hardware and Software Requirement
Core i5; RAM > 4GB; 1 TB Hard Disk; C/C++ and Python on window (32/64 bits) platform

14. List of Experiments

1. OpenCV installation procedure for Windows/Linux   Environment Set-up using Open CV and Programming languages (C, C++, and Python)   Reading and displaying an image
2. Drawing Shapes
3. Image Processing   Enhancing low light image   Image Morphology   Applying Geometric Transforms to an Image
4. Image steganography
5. Segmenting Images and obtaining interesting points
6. Calibration and Stereo Images   Understanding Stereo Images   Generating a Depth Map
7. Edge detection and real-time video processing
8. Exploring structure from motion using OpenCV
9. Number plate recognition using SVM and Neural Network

10. Non-rigid face tracking
11. 3D Head pose estimation using Active Appearance Models (AAM)
12. Recognizing Objects Detecting shapes, Faces and People

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	Introduction to Statistics
7.	Course Basket	Computer Vision & Biometrics

8.   Course Summary

This course will introduce students about 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 able to implement various image enhancement techniques like filtering, object extraction. They will also able to understand the classification of objects through feature extraction.

9.   Course Objectives

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

10.   Course Outcomes

On successful completion of the course, students will be able to achieve the following:
i.   Understand the basis of image processing.
j.   Understand the spatial domain and frequency domain approaches of digital image processing.
k.   Implement various techniques associated with image filtering i.e. smoothing and sharpening.
l.   Learn and implement the image enhancement techniques.
m.   To understand the approaches for enhancement of colour images.
n.   Implement the concepts of classification through object detection followed by feature extraction.

11.   Curriculum Content

Unit-1: Introduction   [5]

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   [4]

Fourier Transform and the Frequency Domain, Basis of Filtering in Frequency Domain, Filters – Low-pass, 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 [5]

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 [6]

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 [6]

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

Text Books

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

Reference Books

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

12.   Teaching and Learning Strategy

The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in Moodle to enable them have appropriate reading.

13.   Hardware and Software Requirement

Core i3/i5; RAM > 4GB; 500 GB/ 1 TB Hard Disk; MATLAB 2019b on Window platform 32/64 bits

14.   List of Experiments

The primary objective of this lab is to supplement an undergraduate level course on image processing and enable students to understand the subject better. The lab consists of a diverse set of experiments with objective, theory, assessment, references and interactive examples which are designed to improve the clarity in understanding of the basic concepts. It is important that the student goes through the objectives and the underlying theory before carrying out the experiments to get maximum benefit.

1. To write a MATLAB program to perform following operations on an image. (a) To read and display an image. (b) To perform data type conversion to 8 bit, 16 bit, 32 bit and double data types. (c) To perform file format conversion in jpg, jpeg, png, bmp and tiff formats
2. To write a MATLAB program to perform following operations. (a) To read and display an image. (b) To create a Gray-scale image and perform masking operation. (c) To perform threshold operation

3. To write a MATLAB program to perform Arithmetic operations (a) Addition (b) Subtraction (c) Multiplication (d) Division
4. To write a MATLAB program to perform following operations. (a) To add the following noises to an image: Salt & pepper, Speckle, Poisson, Gaussian. (b) To filter an image using the following types of filter: Median Weiner Gaussian Laplacian
5. To write a MATLAB program to perform the following morphological operations on the given image: (a) Dilation (b) Erosion (c) Opening (d) Closing
6. To write a MATLAB program to perform the following enhancements on an image: (a) Brightness and Contrast Adjustment (b) Gamma Transformation/ Power Law Transformation (c) Gray-level Slicing (d) Bit Plane Slicing (e) Logarithmic transformation (f) Contrast stretching
7. To write a MATLAB program to perform the detection of edges by following algorithms: (a) Sobel (b) Canny (c) Roberts (d) Prewitt
8. Generate the DFT basis image for the given matrix dimension
9. Find the Discrete Fourier Transform of an image
10. Prove the rotational property & convolution of DFT
11. Generate a Basis images with the given matrix dimension using DCT transforms
12. Perform the discrete cosine transform on the given image
13. Enhance contrasts the given images using histogram equalization

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	Core programming and algorithm skills Basic probability and Statistics
7.	Course Basket	Computer Vision & Biometrics

8. 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.

9. 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.

10. Course Outcomes

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

o. Knowledge and understanding

1) Be familiar with the fundamentals of hypermedia systems, and hypermedia design and usability methodologies.
2) Understand the difficulty of representing and retrieving documents.
3) Understand the latest technologies for linking, describing and searching the Web.
4) Understand the relationship between IR, hypermedia, and semantic models.

p. Cognitive skills (thinking and analysis)

1) Be familiar with classical techniques of Information Retrieval, and the additional techniques employed by Web search engines sufficient to understand how Web search engines work and how they could be improved.
2) Be familiar with techniques for conveying the meaning of documents or hypermedia content, for example, metadata, thesauri, and classification taxonomies
3) Sufficient to understand their application to the "semantic Web".

q. Communication skills (personal and academic)

1) Be familiar with the fundamentals of hypermedia systems sufficient to know how to develop a good Web hypermedia and why a Web site is good or bad.

r. Practical and subject specific skills (Transferable Skills)

1) Be able to implement techniques for the preprocessing needed for information retrieval systems.
2) Be able to develop a small information retrieval system.

11.   Curriculum Content

Unit-1: Introduction to Information Retrieval   [8]
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 lists 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   [6]
Boolean, vector space, TFIDF, Okapi, probabilistic, language modelling, latent semantic indexing. Vector space scoring. The cosine measure. Efficiency considerations. Document length normalization. Relevance feedback and query expansion.

Unit-3: Performance Evaluation   [8]
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 hyper planes; centroids; k Nearest Neighbours. Support vector machine classifiers. Kernel functions. Boosting.

Unit-4: Text Clustering   [7]
Clustering versus classification. Partitioning methods. K-means clustering. Mixture of Gaussians model. Hierarchical agglomerative clustering. Clustering terms using documents.

Unit-5: Web Information Retrieval   [7]
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. 2008. Introduction to Information Retrieval, Cambridge university press

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

12.   Teaching and Learning Strategy
The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in Moodle to enable them have appropriate reading.

13.   **Hardware and Software Requirement:**   Not Applicable

14.   **List of Experiments:**   Not Applicable

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
2.	Course Code	CSF341
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)	
7.	Course Basket	Computer Vision & Biometrics

8.   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.

9.   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.

10.   Course Outcomes

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

s.   Differentiate various learning approaches, and to interpret the concepts of supervised and unsupervised learning.

t.   Compare the different dimensionality reduction techniques.

u.   Illustrate the working of classifier models like SVM, Neural Networks and identify classifier model for typical machine learning applications.

v.   Apply clustering algorithms and identify its applicability in real life problems.

11.   Curriculum Content

Unit-1   [8]

The Machine Learning Landscape: What Is Machine Learning, Why Use Machine Learning, Types of Machine Learning Systems, Supervised/Unsupervised Learning, Batch and Online Learning, Instance-Based Versus Model-Based Learning, Main Challenges of Machine Learning, Testing and Validating.

Unit-2   [9]

Classification: MNIST, Training a Binary Classifier, Performance Measures: Measuring Accuracy Using Cross-Validation, Confusion Matrix, Precision and Recall, Precision/Recall Trade-off, the ROC Curve. Multiclass Classification, Error Analysis, Multilevel Classification, Multi-Output Classification.

Unit-3   [9]

Training Models: Linear Regression, The Normal Equation, And Computational Complexity. Gradient Descent, Batch Gradient Descent, Stochastic Gradient Descent, Mini-batch Gradient Descent. Polynomial Regression, Learning Curves, Regularized Linear Models, Ridge Regression, Lasso Regression, Elastic

Net, Early Stopping, and Logistic Regression, Estimating Probabilities, Training and Cost Function, Decision Boundaries, Softmax Regression.

Text Books

1. Aurélien Géron, Hands on Machine Learning with Scikit-Learn & Tensor Flow, O'Reilly

Reference Books

1. Tom Mitchell, Machine Learning, McGraw Hill, 1997
2. Ethem Alpaydin, Introduction to Machine Learning, PHI

12.   Teaching and Learning Strategy

All materials (ppts, assignments, labs, etc.) will be uploaded in Moodle. Refer to your course in Moodle for details.

13.   Hardware and Software Requirement

Core i5; RAM > 4GB; 1 TB Hard Disk; Python / R Programming Language on window platform

14.   List of Experiments

1. Environment Set-up Download set-up files Installation Environment Set-up Demo Program
2. Data Handling Downloading Dataset Import/Export Dataset Files Summarization of dataset
3. Data Pre-processing Missing Value Outlier Handling Formatting data
4. Descriptive Statistics variance, standard deviation, shape – skewness, kurtosis, percentiles, five point summary boxplots, histograms, bar plot, pie chart, scatter plot, two way tables, covariance, correlation analysis, Chi-Square test for two way tables
5. Supervised Learning- Regressions Linear Regression with one variable Linear Regression with multiple variable Polynomial regression
6. Supervised Learning- Classifications Logistic Regression Decision Tree k-Nearest Neighbours Support Vector Machine
7. Unsupervised Learning- Clustering K-means clustering Hierarchical clustering

8. Deep Learning
Implementation of ANN
Implementation of CNN
Implementation of RNN
9. Data Transfer Learning
Importing pre-trained models
Implementation of pre-trained models with a new learning model

15.   References

a)   IIT Kharagpur (https://cse.iitkgp.ac.in/~saptarshi/courses/ml2018spring/)

b)   VNIT, Nagpur (http://cse.vnit.ac.in/people/rbkeskar/teaching-2/current-courses/machine-learning/)

c)   Illinois Institute of Technology, Chicago (http://www.cs.iit.edu/~agam/cs584/#syllabus)

d)   Massachusetts Institute of Technology, Massachusetts (http://lcsl.mit.edu/courses/ml/1718/)

Undergraduate Course Description Document

1.	Department offering the course	Computer Science and Engineering
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)	
7.	Course Basket	Computer Vision & Biometrics

8.   Course Summary

This course will introduce students about 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 able to implement various image enhancement techniques like filtering, object extraction. They will also able to understand the classification of objects through feature extraction.

9.   Course Objectives

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

10.   Course Outcomes

On successful completion of the course, students will be able to achieve the following:
w.   Select the type of remote sensing technique / data for required purpose.
x.   Identify the earth surface features from satellite images.
y.   Analyse the energy interactions in the atmosphere and earth surface features.
z.   Get familiarized about various image enhancement and image processing techniques.
aa.   Interpret images and prepare thematic maps.

11.   Curriculum Content

Unit-1: Fundamentals   [5]

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   [5]

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 – opto-mechanical scanners – pushbroom and whiskbroom cameras – Panchromatic, multi spectral, hyperspectral scanners – geometric characteristics of scanner, imagery - Landsat, SPOT, IRS, World View.

Unit-3: Image Enhancements **[5]**

Spectral signatures – Image characteristics, feature space scatterogram- point, local and regional operation – spatial feature and multi-image manipulation techniques - principle component analysis - Optimal Rotation Transformation – Scale-space transform, 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 **[5]**

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 - sub-pixel and super-pixel classification.

Unit-5: Image Analysis and Change Detection **[6]**

Pattern recognition - boundary detection and representation - textural and contextual analysis - decision concepts: Fuzzy sets - evidential reasoning - Hyper-spectral 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

Text Books

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.

Reference Books

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

12.   Teaching and Learning Strategy

The teaching of students will be conducted through power point lectures, tutorials, short classroom exercises aimed at solving real life problems. The lecture material will be availed to the students in LMS to enable them have appropriate reading.

13.   Hardware and Software Requirement

Core i5; RAM > 4GB; 1 TB Hard Disk; ILWIS 3.3 on window platform (32/64 bits)

14.   List of Experiments

1. Demonstration of single and multivariate satellite image
2. Reading and Generating False Colour Composite (FCC); Extracting area of Interest (AOI)
3. Generating Histogram of various bands
4. Geo-referencing the base image
5. Geometric correction of satellite image
6. Enhancement using different techniques

7. Preparation of Land use/land cover map using Satellite Data
8. Classification using Neural Network and Fuzzy Logic
9. Accuracy Assessment and Change detection study
10. Introduction to Mat-Lab toolbox for GIS
11. Extraction of various features
12. Preparation and analysis of spectral signatures using handheld spectro-radiometer for Vegetation, Soil, Water, Stone/Rock